

世纪星铣床数控系统

HNC-18iM/19iM

编程说明书

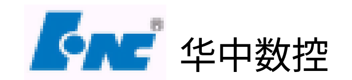


V1.2

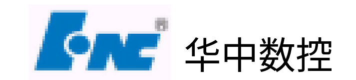
2006.02

武汉华中数控股份有限公司

中国·武汉



目　　录


华中数控


华中数控



第一章 概述

本章的目的是对本说明书中提到的一些基本概念进行解释。

1.1 数控编程概述

1.1.1 定义零件程序

零件程序是由数控装置专用编程语言书写的一系列指令组成的（应用得最广泛的是 ISO 码：国际标准化组织规定的代码）。]

数控装置将零件程序转化为对机床的控制动作。

最常使用的程序存储介质是穿孔纸带和磁盘。

1.1.2 准备零件程序

如图 1.1.1 所示，可以用传统的方法手工编制一个零件程序，也可以用一套 CAD/CAM 系统（如目前流行的 MasterCAM 系统）来创建一个零件程序。

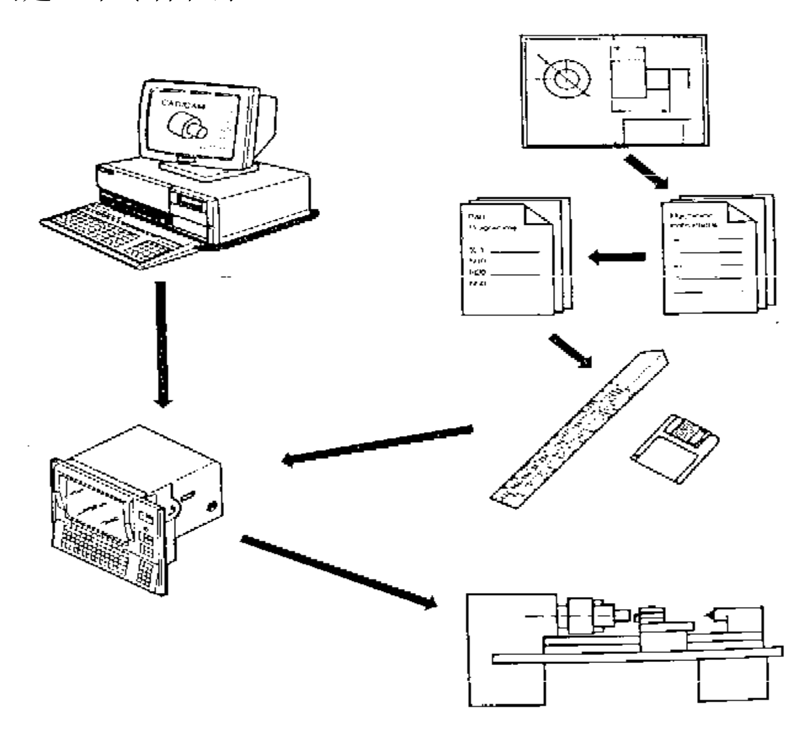


图 1.1.1　准备一个零件程序

1.2 数控机床概述

1.2.1 机床坐标轴

为简化编程和保证程序的通用性，对数控机床的坐标轴和方向命



名制订了统一的标准，规定直线进给坐标轴用 *X*，*Y*，*Z* 表示，常称基本坐标轴。*X*，*Y*，*Z* 坐标轴的相互关系用右手定则决定，如图 1.2.1 所示，图中大姆指的指向为 *X* 轴的正方向，食指指向为 *Y* 轴的正方向，中指指向为 *Z* 轴的正方向。

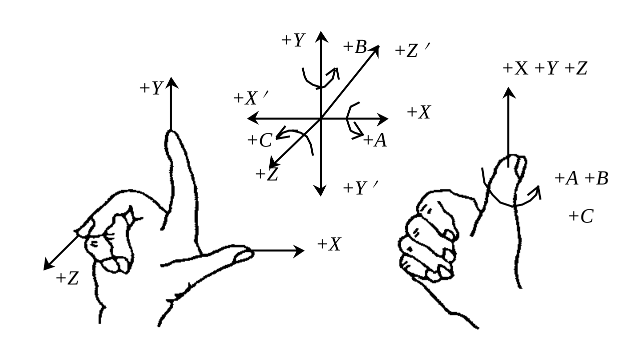


图 1.2.1　机床坐标轴

围绕 *X*，*Y*，*Z* 轴旋转的圆周进给坐标轴分别用 *A*，*B*，*C* 表示，根据右手螺旋定则，如图所示，以大姆指指向 +*X*，+*Y*，+*Z* 方向，则食指、中指等的指向是圆周进给运动的 +*A*，+*B*，+*C* 方向。

数控机床的进给运动，有的由主轴带动刀具运动来实现，有的由工作台带着工件运动来实现。上述坐标轴正方向，是假定工件不动，刀具相对于工件做进给运动的方向。如果是工件移动则用加“′”的字母表示，按相对运动的关系，工件运动的正方向恰好与刀具运动的正方向相反，即有：

$$+X = -X' ,\quad +Y = -Y' ,\quad +Z = -Z' ,$$

$$+A = -A' ,\quad +B = -B' ,\quad +C = -C'$$

同样两者运动的负方向也彼此相反。

机床坐标轴的方向取决于机床的类型和各组成部分的布局，对铣床而言：

——*Z* 轴与主轴轴线重合，刀具远离工件的方向为正方向(+*Z*)；

——*X* 轴垂直于 *Z* 轴，并平行于工件的装卡面，如果为单立柱铣床，面对刀具主轴向立柱方向看，其右运动的方向为 *X* 轴的正方向(+*X*)；

——*Y* 轴与 *X* 轴和 *Z* 轴一起构成遵循右手定则的坐标系统。

1.2.2 机床坐标系、机床零点和机床参考点

机床坐标系是机床固有的坐标系，机床坐标系的原点也称为机床原点或机床零点。在机床经过设计、制造和调整后，这个原点便



被确定下来，它是固定的点。

数控装置上电时并不知道机床零点，每个坐标轴的机械行程是由最大和最小限位开关来限定的。

为了正确地在机床工作时建立机床坐标系，通常在每个坐标轴的移动范围内设置一个机床参考点（测量起点），机床起动时，通常要进行机动或手动回参考点，以建立机床坐标系。

机床参考点可以与机床零点重合，也可以不重合，通过参数指定机床参考点到机床零点的距离。

机床回到了参考点位置，也就知道了该坐标轴的零点位置，找到所有坐标轴的参考点，CNC 就建立起了机床坐标系。

机床坐标轴的有效行程范围是由软件限位来界定的，其值由制造商定义。机床零点（OM）、机床参考点（Om）、机床坐标轴的机械行程及有效行程的关系如图 1.2.2 所示。

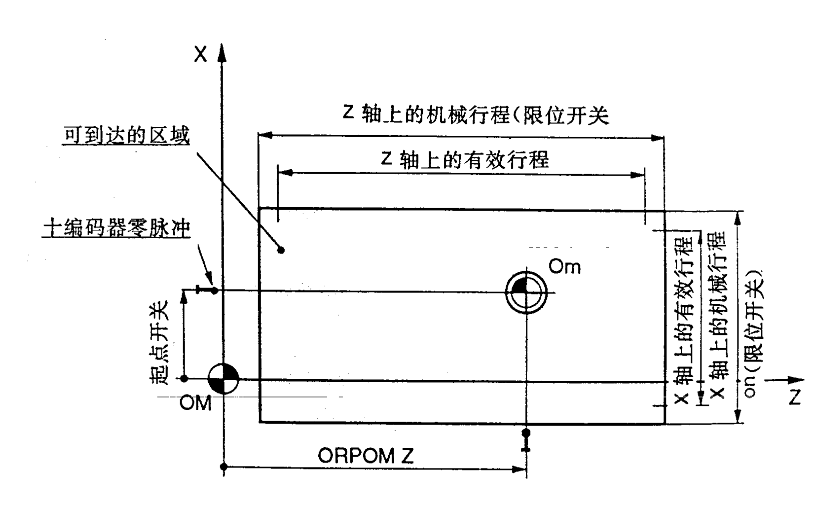
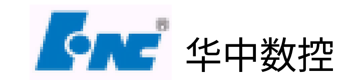


图 1.2.2　机床零点 OM 和机床参考点 Om

1.2.3 工件坐标系、程序原点和对刀点

工件坐标系是编程人员在编程时使用的，编程人员选择工件上的某一已知点为原点（也称程序原点），建立一个新的坐标系，称为工件坐标系。工件坐标系一旦建立便一直有效，直到被新的工件坐标系所取代。

工件坐标系的原点选择要尽量满足编程简单，尺寸换算少，引起的加工误差小等条件。一般情况下，以坐标式尺寸标注的零件，程序原点应选在尺寸标注的基准点；对称零件或以同心圆为主的零件，程序原点应选在对称中心线或圆心上。Z 轴的程序原点通常选在工件的上表面。



对刀点是零件程序加工的起始点，对刀的目的是确定程序原点在机床坐标系中的位置，对刀点可与程序原点重合，也可在任何便于对刀之处，但该点与程序原点之间必须有确定的坐标联系。
可以通过 CNC 将相对于程序原点的任意点的坐标转换为相对于机床零点的坐标。
加工开始时要设置工件坐标系，用 G92 指令可建立工件坐标系；用 G54~G59 指令可选择工件坐标系。



1. 第二章 零件程序的结构

一个零件程序是一组被传送到数控装置中去的指令和数据。

一个零件程序是由遵循一定结构、句法和格式规则的若干个程序段组成的，而每个程序段是由若干个指令字组成的。如图 2.1 所示。

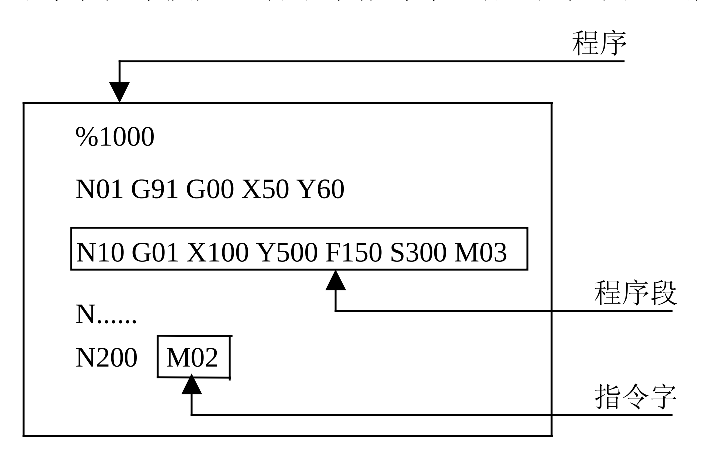


图 2. 1 程序的结构

2.1 指令字的格式

一个指令字是由地址符(指令字符)和带符号（如定义尺寸的字）或不带符号（如准备功能字 G 代码）的数字数据组成的。

程序段中不同的指令字符及其后续数值确定了每个指令字的含义。在数控程序段中包含的主要指令字符如表 2.1 所示。

表 2.1 指令字符一览表

机 能	地 址	意 义
零件程序号	%	程序编号：%1~4294967295
程序段号	N	程序段编号：N0~4294967295
准备机能	G	指令动作方式(直线、圆弧等) G00-99
尺寸字	X，Y，Z A，B，C U，V，W	坐标轴的移动命令±99999.999
	R	圆弧的半径，固定循环的参数
	I，J，K	圆心相对于起点的坐标，固定循环的参数
进给速度	F	进给速度的指定 F0~24000
主轴机能	S	主轴旋转速度的指定 S0~9999
刀具机能	T	刀具编号的指定 T0~99



辅助机能	M	机床侧开/关控制的指定　M0~99
补偿号	H，D	刀具补偿号的指定　00~99
暂停	P，X	暂停时间的指定　秒
程序号的指定	P	子程序号的指定　P1~4294967295
重复次数	L	子程序的重复次数，固定循环的重复次数
参数	P，Q，R	固定循环的参数

2.2 程序段的格式

一个程序段定义一个将由数控装置执行的指令行。

程序段的格式定义了每个程序段中功能字的句法，如图 2.2.1 所示。

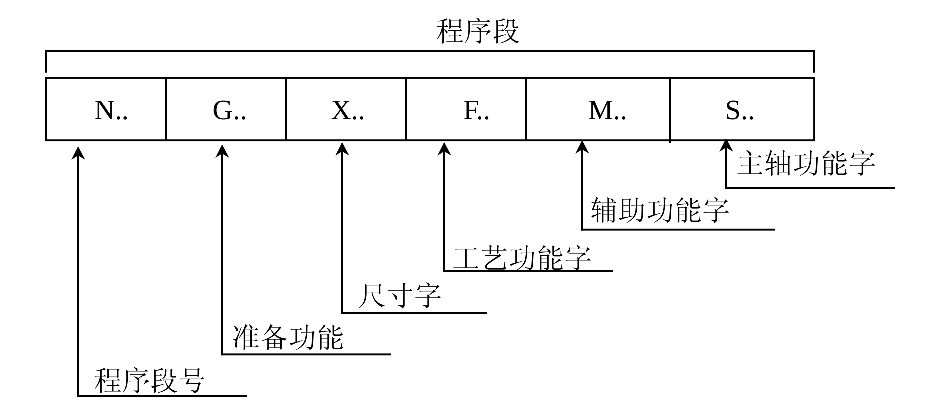


图 2.2.1 程序段格式

2.3 程序的一般结构

一个零件程序必须包括起始符和结束符。

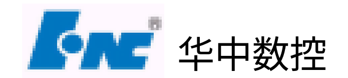
一个零件程序是按程序段的输入顺序执行的，而不是按程序段号的顺序执行的，但书写程序时，建议按升序书写程序段号。

华中世纪星数控装置 HNC-18iM/19iM 的程序结构：

程序起始符：%(或 O)符，%(或 O)后跟程序号

程序结束：M02 或 M30。

注释符：括号( )内或分号；后的内容为注释文字



2. 第三章 HNC-18iM/19iM 的编程指令体系

3.1 辅助功能 M 代码

辅助功能由地址字 M 和其后的一或两位数字组成，主要用于控制零件程序的走向，以及机床各种辅助功能的开关动作。

M 功能有**非模态 M 功能和模态 M 功能**二种形式。

- 非模态 M 功能 (当段有效代码) ：只在书写了该代码的程序段中有效；
- 模态 M 功能(续效代码)：一组可相互注销的 M 功能，这些功能在被同一组的另一个功能注销前一直有效。

模态 M 功能组中包含一个缺省功能（见表 3.1），系统上电时将被初始化为该功能。

另外，M 功能还可分为**前作用 M 功能和后作用 M 功能**二类。

- 前作用 M 功能：在程序段编制的轴运动之前执行；
- 后作用 M 功能：在程序段编制的轴运动之后执行。

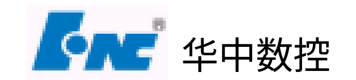
华中世纪星 HNC-18iM/19iM 数控装置 M 指令功能如表 3.1 所示（▶标记者为缺省值）：

表 3.1 M 代码及功能

代 码	模态	功 能 说 明	代 码	模态	功 能 说 明
M00	非模态	程序停止	M03	模态	主轴正转起动
M02	非模态	程序结束	M04	模态	主轴反转起动
M30	非模态	程序结束并返回程序起点	M05	模态	▶ 主轴停止转动
			M06	非模态	换刀
M98	非模态	调用子程序	M07	模态	切削液打开
M99	非模态	子程序结束	M09	模态	▶ 切削液停止

其中：

- M00、M02、M30、M98、M99 用于控制零件程序的走向，是 CNC 内定的辅助功能，不由机床制造商设计决定，也就是说，与 PLC 程序无关；
- 其余 M 代码用于机床各种辅助功能的开关动作，其功能不由 CNC 内定，而是由 PLC 程序指定，所以有可能因机床制造厂不同而有差异(表内为标准 PLC 指定的功能)，请使用者参考机床说明书。



3.1.1 CNC 内定的辅助功能

(1) 程序暂停 M00

当 CNC 执行到 M00 指令时，将暂停执行当前程序，以方便操作者进行刀具和工件的尺寸测量、工件调头、手动变速等操作。
暂停时，机床的主轴、进给及冷却液停止，而全部现存的模态信息保持不变，欲继续执行后续程序，重按操作面板上的“循环启动”键。
M00 为非模态后作用 M 功能。

(2) 程序结束 M02

M02 编在主程序的最后一个程序段中。
当 CNC 执行到 M02 指令时，机床的主轴、进给、冷却液全部停止，加工结束。
使用 M02 的程序结束后，若要重新执行该程序，就得重新调用该程序，或在自动加工子菜单下，按 F4 键（请参考 HNC-18iM/19iM 操作说明书），然后再按操作面板上的“循环启动”键。
M02 为非模态后作用 M 功能。。

(3) 程序结束并返回到零件程序头 M30

M30 和 M02 功能基本相同，只是 M30 指令还兼有控制返回到零件程序头(%)的作用。
使用 M30 的程序结束后，若要重新执行该程序，只需再次按操作面板上的“循环启动”键。

(4) 子程序调用 M98 及从子程序返回 M99

M98 用来调用子程序。
M99 表示子程序结束，执行 M99 使控制返回到主程序。
(i) 子程序的格式

```
%****
……
M99
```

在子程序开头，必须规定子程序号，以作为调用入口地址。在子程序的结尾用 M99，以控制执行完该子程序后返回主程序。
(ii) 调用子程序的格式

M98 P_ L_

P：被调用的子程序号
L：重复调用次数

注：可以带参数调用子程序，请参考附录 1。

G65 指令的功能和参数与 M98 相同。

3.1.2 PLC 设定的辅助功能

(1) 主轴控制指令 M03、M04、M05

M03 启动主轴以程序中编制的主轴速度顺时针方向（从 Z 轴正向朝 Z 轴负向看）旋转。

M04 启动主轴以程序中编制的主轴速度逆时针方向旋转。

M05 使主轴停止旋转。

M03、M04 为模态前作用 M 功能；M05 为模态后作用 M 功能，M05 为缺省功能。

M03、M04、M05 可相互注销。

(2) 换刀指令 M06

M06 用于在加工中心上调用一个欲安装在主轴上的刀具。

刀具将被自动地安装在主轴上。

M06 为非模态后作用 M 功能。

(3) 冷却液打开、停止指令 M07、M09

M07 指令将打开冷却液管道。

M09 指令将关闭冷却液管道。

M07 为模态前作用 M 功能；M09 为模态后作用 M 功能，M09 为缺省功能。

3.2 主轴功能 S、进给功能 F 和刀具功能 T

3.2.1 主轴功能 S

主轴功能 S 控制主轴转速，其后的数值表示主轴速度，单位为转/每分钟(r/min)。

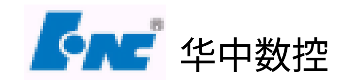
S 是模态指令，S 功能只有在主轴速度可调节时有效。

3.2.2 进给速度 F

F 指令表示工件被加工时刀具相对于工件的合成进给速度，F 的单位取决于 G94(每分钟进给量 mm/min)或 G95(每转进给量 mm/r)。

当工作在 G01，G02 或 G03 方式下，编程的 F 一直有效，直到被新的 F 值所取代，而工作在 G00、G60 方式下，快速定位的速度是各轴的最高速度，与所编 F 无关。

借助操作面板上的倍率按键，F 可在一定范围内进行倍率修调。



当执行攻丝循环 G84，螺纹切削 G33 时，倍率开关失效，进给倍率固定在 100％。

3.2.3 刀具功能(T 机能)

T 代码用于选刀，其后的数值表示选择的刀具号，T 代码与刀具的关系是由机床制造厂规定的。

在加工中心上执行 T 指令，刀库转动选择所需的刀具，然后等待，直到 M06 指令作用时自动完成换刀。

T 指令同时调入刀补寄存器中的刀补值(刀补长度和刀补半径)。T 指令为非模态指令，但被调用的刀补值一直有效，直到再次换刀调入新的刀补值。

3.3 准备功能 G 代码

准备功能 G 指令由 G 后一或二位数值组成，它用来规定刀具和工件的相对运动轨迹、机床坐标系、坐标平面、刀具补偿、坐标偏置等多种加工操作。

华中世纪星 HNC-18iM/19iM 数控装置 G 功能指令见附表 1

G 功能有**非模态 G 功能和模态 G 功能**之分。

- 非模态 G 功能：只在所规定的程序段中有效，程序段结束时被注销；
- 模态 G 功能：一组可相互注销的 G 功能，这些功能一旦被执行，则一直有效，直到被同一组的 G 功能注销为止。

模态 G 功能组中包含一个缺省 G 功能（附表 1 中有标记者），▶上电时将被初始化为该功能。

没有共同参数的不同组 G 代码可以放在同一程序段中，而且与顺序无关。例如，G90、G17 可与 G01 放在同一程序段，但 G24、G68、G51 等不能与 G01 放在同一程序段。

3.3.1 有关单位的设定

(1) 尺寸单位选择 G20，G21，G22

格式：G20

G21

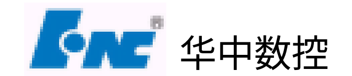
G22

说明：

G20：英制输入制式；

G21：公制输入制式；

G22：脉冲当量输入制式。



3 种制式下线性轴、旋转轴的尺寸单位如表 3.2 所示。
G20、G21、G22 为模态功能，可相互注销，G21 为缺省值。

表 3.2　尺寸输入制式及其单位

	线性轴	旋转轴
英制(G20)	英寸	度
公制(G21)	毫米	度
脉冲当量(G22)	移动轴脉冲当量	旋转轴脉冲当量

(2) 进给速度单位的设定 G94、G95

格式： G94 [F_]；
G95 [F_]；

说明：

G94：每分钟进给；
G95：每转进给。

G94 为每分钟进给。对于线性轴，F 的单位依 G20/G21/G22 的设定而为 mm/min，in/min 或脉冲当量/min；对于旋转轴，F 的单位为度/min 或脉冲当量/min。

G95 为每转进给，即主轴转一周时刀具的进给量。F 的单位依 G20/G21/G22 的设定而为 mm/r，in/r 或脉冲当量/r 。这个功能只在主轴装有编码器时才能使用。

G94、G95 为模态功能，可相互注销，G94 为缺省值。

3.3.2 有关坐标系和坐标的指令

(1) 绝对值编程 G90 与相对值编程 G91

格式： G90
G91

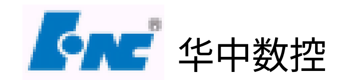
说明：

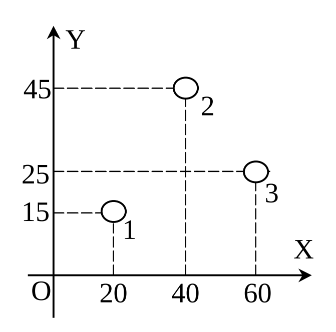
G90：绝对值编程，每个编程坐标轴上的编程值是相对于程序原点的。
G91：相对值编程，每个编程坐标轴上的编程值是相对于前一位置而言的，该值等于沿轴移动的距离。

G90、G91 为模态功能，可相互注销，G90 为缺省值。

G90、G91 可用于同一程序段中，但要注意其顺序所造成的差异。

例 1．如图 3.3.1 所示，使用 G90、G91 编程：要求刀具由原点按顺序移动到 1、2、3 点。





G90 编程

N	X	Y
N01	X20	Y15
N02	X40	Y45
N03	X60	Y25

G91 编程

N	X	Y
N01	X20	Y15
N02	X20	Y30
N03	X20	Y-20

图 3.3.1 G90/G91 编程

选择合适的编程方式可使编程简化。当图纸尺寸由一个固定基准给定时，采用绝对方式编程较为方便；而当图纸尺寸是以轮廓顶点之间的间距给出时，采用相对方式编程较为方便。

(2) 工件坐标系设定 G92

格式：G92 X_Y_Z_A _

说明：

X、*Y*、*Z*、*A*：设定的工件坐标系原点到刀具起点的有向距离(注意：HNC –21M的最大联动轴数为4。本说明书中，假设第四轴用 *A* 表示)。

G92 指令通过设定刀具起点（对刀点）与坐标系原点的相对位置建立工件坐标系。工件坐标系一旦建立，绝对值编程时的指令值就是在此坐标系中的坐标值。

例 2．使用 G92 编程，建立如图 3.3.2 所示的工件坐标系。

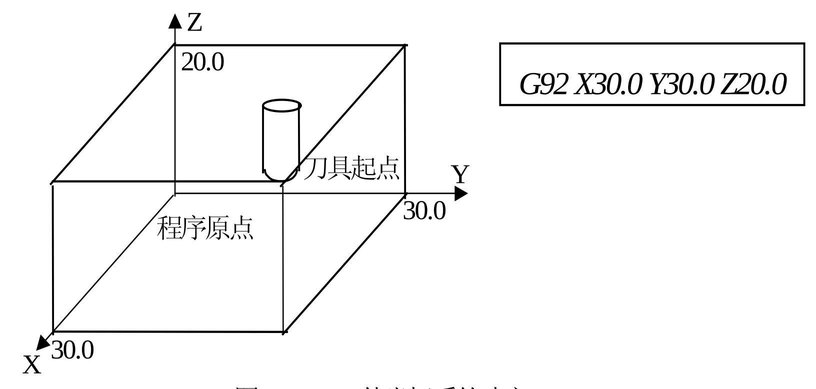


图3.3.2 工件坐标系的建立

执行此程序段只建立工件坐标系，刀具并不产生运动。

G92 指令为非模态指令，一般放在一个零件程序的第一段。

(3) 工件坐标系选择 G54~G59

$$
格式：\left\{\begin{array}{l} G54 \\ G55 \\ G56 \\ G57 \\ G58 \\ G59 \end{array}\right\}
$$

说明：

G54~G59 是系统预定的 6 个工件坐标系(如图 3.3.3)，可根据需要任意选用。

这 6 个预定工件坐标系的原点在机床坐标系中的值(工件零点偏置值)可用 MDI 方式输入，系统自动记忆。

工件坐标系一旦选定，后续程序段中绝对值编程时的指令值均为相对此工件坐标系原点的值。

G54~G59 为模态功能，可相互注销，G54 为缺省值。

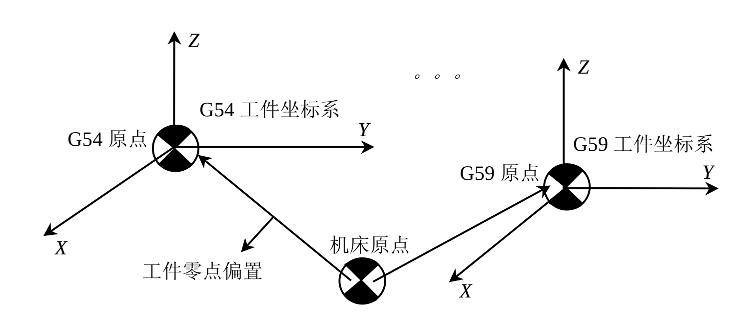
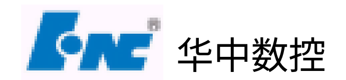


图 3.3.3 工件坐标系选择(G54~G59)

例 3．如图 3.3.4 所示，使用工件坐标系编程：要求刀具从当前点移动到 A 点，再从 A 点移动到 B 点。



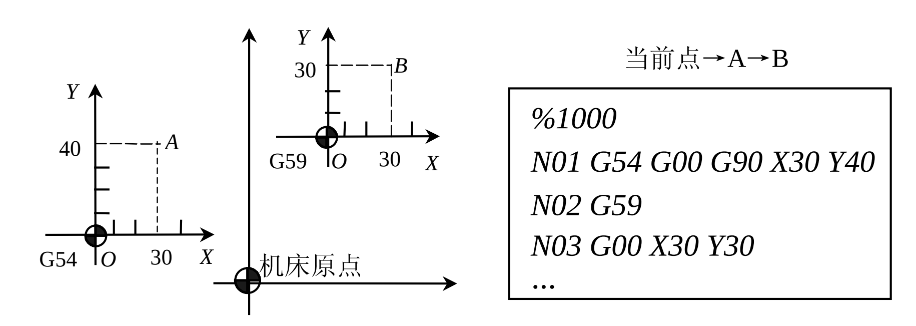


图 3.3.4 使用工件坐标系编程

注意：

使用该组指令前，先用 MDI 方式输入各坐标系的坐标原点在机床坐标系中的坐标值。

(4) 局部坐标系设定 G52

格式：G52 X_Y_Z_A_

说明：

X、*Y*、*Z*、*A*：局部坐标系原点在当前工件坐标系中的坐标值。

G52 指令能在所有的工件坐标系(G92、G54~G59)内形成子坐标系，即局部坐标系，如图 3.3.5。

含有 G52 指令的程序段中，绝对值编程方式的指令值就是在该局部坐标系中的坐标值。

设定局部坐标系后，工件坐标系和机床坐标系保持不变。

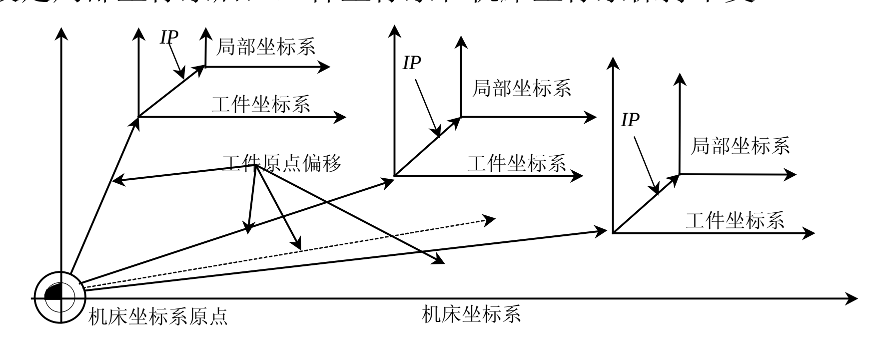
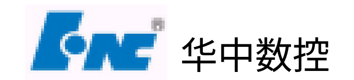


图 3.3.5　局部坐标系的设定 G52

G52 指令为非模态指令。

在缩放及旋转功能下，不能使用 G52 指令，但在 G52 下能进行缩放及坐标系旋转。



(5) 直接机床坐标系编程 G53

格式：G53

说明：

G53 是机床坐标系编程，在含有 G53 的程序段中，绝对值编程时的指令值是在机床坐标系中的坐标值。

G53 指令为非模态指令。

(6) 坐标平面选择 G17，G18，G19

格式： G17

G18

G19

说明：

G17：选择 *XY* 平面；

G18：选择 *ZX* 平面；

G19：选择 *YZ* 平面。

该组指令选择进行圆弧插补和刀具半径补偿的平面。

G17、G18、G19 为模态功能，可相互注销，G17 为缺省值。

注意：

移动指令与平面选择无关。例如指令

G17 G01 Z10

时，*Z* 轴照样会移动。

3.3.3 进给控制指令

(1) 快速定位 G00

格式：G00 X_Y_Z_A_

说明：

X、*Y*、*Z*、*A*：快速定位终点，在 G90 时为终点在工件坐标系中的坐标；在 G91 时为终点相对于起点的位移量。

G00 指令刀具相对于工件以各轴预先设定的速度，从当前位置快速移动到程序段指令的定位目标点。

G00 指令中的快移速度由机床参数“快移进给速度”对各轴分别设定，不能用 F__规定。

G00 一般用于加工前快速定位或加工后快速退刀。

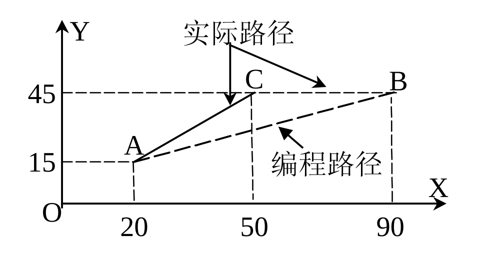
快移速度可由面板上的快速修调旋钮修正。

G00 为模态功能，可由 G01、G02、G03 或 G33 功能注销。

注意：

在执行 G00 指令时，由于各轴以各自速度移动，不能保证各轴同时到达终点，因而联动直线轴的合成轨迹不一定是直线。操作者必须格外小心，以免刀具与工件发生碰撞。常见的做法是，将 Z 轴移动到安全高度，再放心地执行 G00 指令。

例 4．如图 3.3.6 所示，使用 G00 编程：要求刀具从 A 点快速定位到 B 点。



从 A 到 B 快速定位

绝对值编程：

G90 G00 X90 Y45

增量值编程:

G91 G00 X70 Y30

图 3.3.6　G00 编程

当 X 轴和 Y 轴的快进速度相同时，从 A 点到 B 点的快速定位路线为 A→C→B，即以折线的方式到达 B 点，而不是以直线方式从 A →B。

(2) 单方向定位 G60

格式：G60　X_Y_Z_A_

说明：

X、*Y*、*Z*、*A*：单向定位终点，在 G90 时为终点在工件坐标系中的坐标；在 G91 时为终点相对于起点的位移量。

G60 单方向定位过程：各轴先以 G00 速度快速定位到一中间点，然后以一固定速度移动到定位终点。

各轴的定位方向(从中间点到定位终点的方向)以及中间点与定位终点的距离由机床参数“单向定位偏移值”设定。当该参数值<0 时，定位方向为负，当该参数值>0 时，定位方向为正。

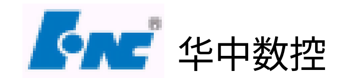
G60 指令仅在其被规定的程序段中有效。

(3) 线性进给 G01

格式：　G01　X _Y_Z_A_F_；

说明：

X、*Y*、*Z*、*A*：线性进给终点，在 G90 时为终点在工件坐标系中的坐标；在 G91 时为终点相对于起



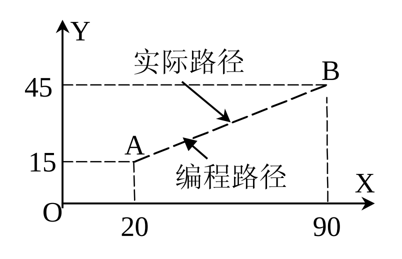
点的位移量；

*F_*：合成进给速度。

G01 指令刀具以联动的方式，按 F 规定的合成进给速度，从当前位置按线性路线(联动直线轴的合成轨迹为直线)移动到程序段指令的终点。

G01 是模态代码，可由 G00、G02、G03 或 G33 功能注销。

例 5．如图 3.3.7 所示，使用 G01 编程：要求从 A 点线性进给到 B 点（此时的进给路线是从 A→B 的直线）。



从 A 到 B 线性进给

绝对值编程：
G90 G01 X90 Y45 F800
增量值编程:
G91 G01 X70 Y30 F800

图 3.3.7　G01 编程

(4) 圆弧进给 G02/G03

格式：$G17\begin{Bmatrix}G02\\G03\end{Bmatrix}X_Y_\begin{Bmatrix}I_J_\\R_\end{Bmatrix}F_$

$G18\begin{Bmatrix}G02\\G03\end{Bmatrix}X_Z_\begin{Bmatrix}I_K_\\R_\end{Bmatrix}F_$

$G19\begin{Bmatrix}G02\\G03\end{Bmatrix}Y_Z_\begin{Bmatrix}J_K_\\R_\end{Bmatrix}F_$

说明：

G02：　顺时针圆弧插补(如图 3.3.8 所示)；

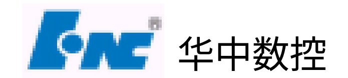
G03：　逆时针圆弧插补(如图 3.3.8 所示)；

G17：　*XY* 平面的圆弧；

G18：　*ZX* 平面的圆弧；

G19：　*YZ* 平面的圆弧。

X, Y, Z：圆弧终点，在 G90 时为圆弧终点在工件坐标系中的坐标；在 G91 时为圆弧终点相对于圆弧起点的位移量；



I, J, K：圆心相对于圆弧起点的偏移值(等于圆心的坐标减去圆弧起点的坐标，如图 3.3.9 所示) ，在 G90/G91 时都是以增量方式指定；

R：圆弧半径，当圆弧圆心角小于 180° 时，*R* 为正值，否则 *R* 为负值

F：被编程的两个轴的合成进给速度。

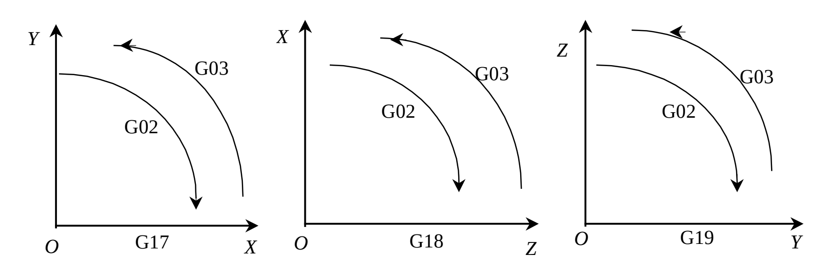


图 3.3.8 不同平面的 G02 与 G03 选择

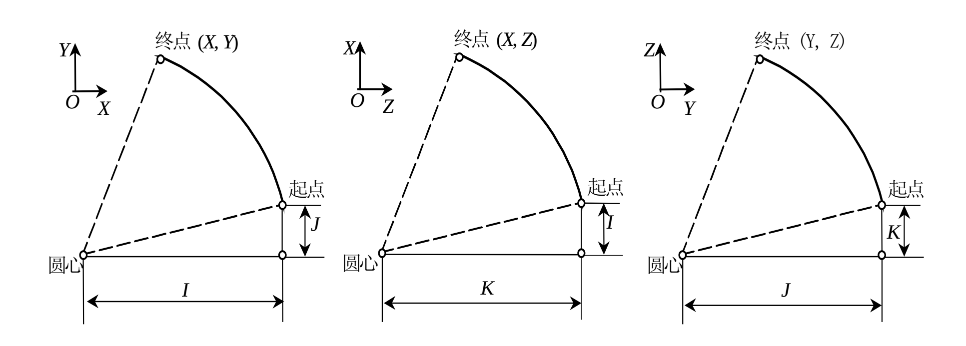


图3.3.9 *I*、*J*、*K* 的选择

例 6 使用 G02 对图 3.3.10 所示劣弧 a 和优弧 b 编程。

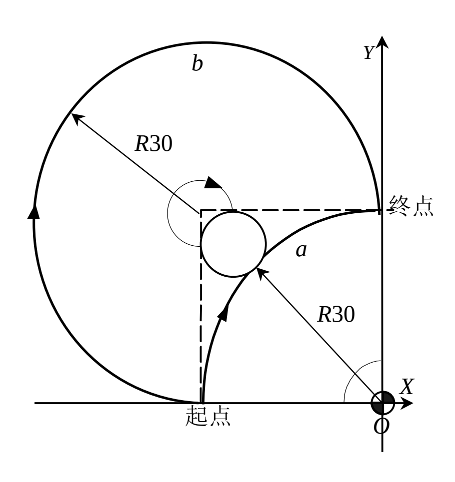
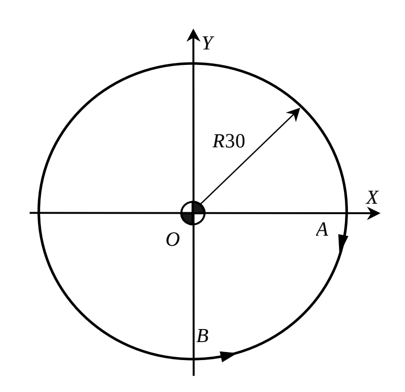


图 3.3.10 圆弧编程

圆弧编程的 4 种方法组合

(i) 圆弧 *a*
G91 G02 X30 Y30 R30 F300
G91 G02 X30 Y30 I30 J0 F300
G90 G02 X0 Y30 R30 F300
G90 G02 X0 Y30 I30 J0 F300
(ii) 圆弧 *b*
G91 G02 X30 Y30 R–30 F300
G91 G02 X30 Y30 I0 J30 F300
G90 G02 X0 Y30 R–30 F300
G90 G02 X0 Y30 I0 J30 F300

例 7 使用 G02/G03 对图 3.3.11 所示的整圆编程。



i）从 *A* 点顺时针一周时
G90 G02 X30 Y0 I–30 J0 F300
G91 G02 X0 Y0 I–30 J0 F300

(ii) 从 *B* 点逆时针一周时
G90 G03 X0 Y–30 I0 J30 F300
G91 G03 X0 Y0 I0 J30 F300

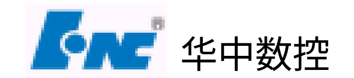
图 3.3.11 整圆编程

注意：

(1) 顺时针或逆时针是从垂直于圆弧所在平面的坐标轴的正方向看到的回转方向；
(2) 整圆编程时不可以使用 *R*，只能用 *I*、*J*，*K*；
(3) 同时编入 *R* 与 *I*、*J*，*K* 时，*R* 有效。

(5) 螺旋线进给 G02/G03

格式：$G17\begin{Bmatrix}G02\\G03\end{Bmatrix}X_Y_\begin{Bmatrix}I_J_\\R_\end{Bmatrix}Z_F_$



$$G18\begin{Bmatrix}G02\\G03\end{Bmatrix}X_Z_\begin{Bmatrix}I_K_\\R_\end{Bmatrix}Y_F_$$

$$G19\begin{Bmatrix}G02\\G03\end{Bmatrix}Y_Z_\begin{Bmatrix}J_K_\\R_\end{Bmatrix}X_F_$$

说明：

X, Y, Z 中由 G17/G18/G19 平面选定的两个坐标为螺旋线投影圆弧的终点，意义同圆弧进给，第 3 坐标是与选定平面相垂直的轴终点；

其余参数的意义同圆弧进给。

该指令对另一个不在圆弧平面上的坐标轴施加运动指令，对于任何小于 360° 的圆弧，可附加任一数值的单轴指令。

例 8 使用 G03 对图 3.3.12 所示的的螺旋线编程。

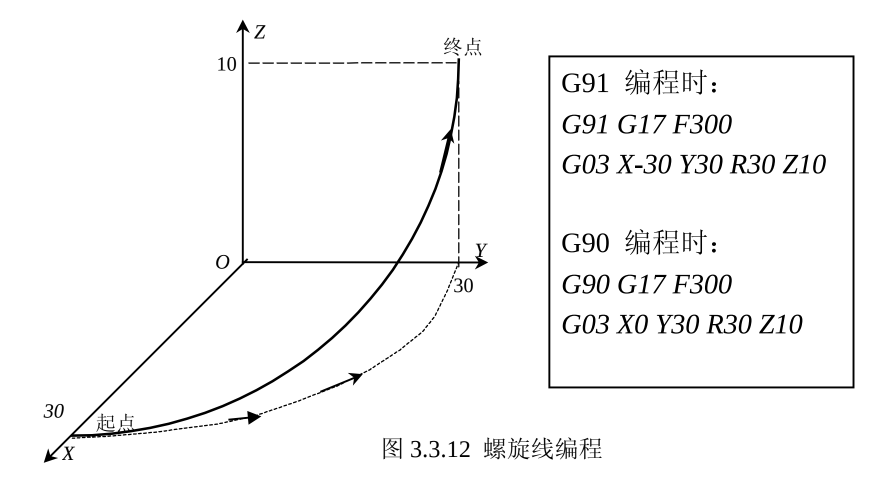


图 3.3.12 螺旋线编程

(6) 虚轴指定 G07 及正弦线插补

格式：G07 X_Y_Z_A_

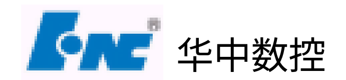
说明：

X、*Y*、*Z*、*A*：被指令轴后跟数字 0，则该轴为虚轴，后跟数字 1，则该轴为实轴。

G07 为虚轴指定和取消指令。G07 为模态指令。

若一轴为虚轴，则此轴只参加计算，不运动。虚轴仅对自动操作有效，对手动操作无效。

用 G07 可进行正弦曲线插补，即在螺旋线插补前，将参加圆弧插补的某一轴指定为虚轴，则螺旋线插补变为正弦线插补。



例 9：使用 G03 对图 3.3.13 所示的正弦线编程。

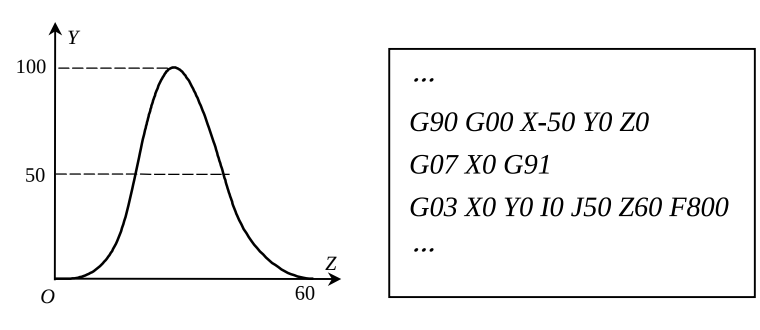


图 3.3.13 正弦线插补编程

3. 3.4 回参考点控制指令

(1) 自动返回参考点 G28

格式：G28 X_Y_Z_A_

说明：

X、*Y*、*Z*、*A*：回参考点时经过的中间点（非参考点），在 G90 时为中间点在工件坐标系中的坐标；在 G91 时为中间点相对于起点的位移量。

G28 指令首先使所有的编程轴都快速定位到中间点，然后再从中间点返回到参考点。

一般，G28 指令用于刀具自动更换或者消除机械误差，在执行该指令之前应取消刀具半径补偿和刀具长度补偿。

在 G28 的程序段中不仅产生坐标轴移动指令，而且记忆了中间点坐标值，以供 G29 使用。

电源接通后，在没有手动返回参考点的状态下，指定 G28 时，从中间点自动返回参考点，与手动返回参考点相同。这时从中间点到参考点的方向就是机床参数“回参考点方向”设定的方向。

G28 指令仅在其被规定的程序段中有效。

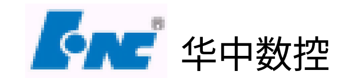
(2) 自动从参考点返回 G29

格式：G29 X _Y_Z_A_

说明：

X、*Y*、*Z*、*A*：返回的定位终点，在 G90 时为定位终点在工件坐标系中的坐标；在 G91 时为定位终点相对于 G28 中间点的位移量。

G29 可使所有编程轴以快速进给经过由 G28 指令定义的中间点，



然后再到达指定点。通常该指令紧跟在 G28 指令之后。
G29 指令仅在其被规定的程序段中有效。

例 10：用 G28、G29 对图 3.3.14 所示的路径编程：要求由 A 经过中间点 B 并返回参考点，然后从参考点经由中间点 B 返回到 C，并在 C 点换刀。

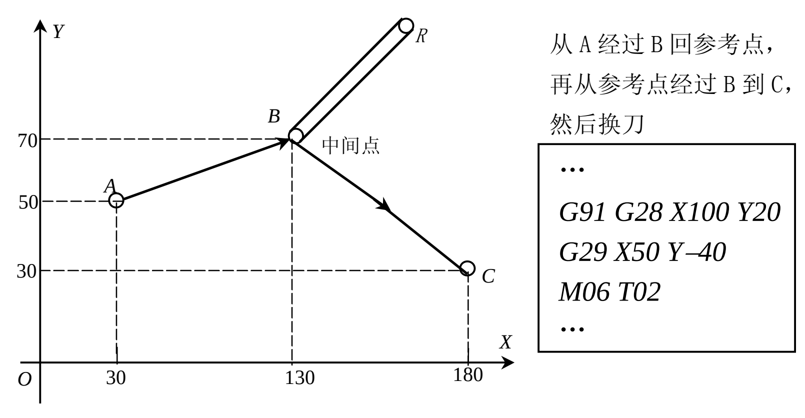


图 3.3.14 G28/G29 编程

本例表明，编程员不必计算从中间点到参考点的实际距离。

3.3.5 刀具补偿功能指令

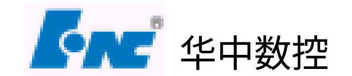
(1) 刀具半径补偿 G40，G41，G42

格式：$\begin{Bmatrix} G17 \\ G18 \\ G19 \end{Bmatrix} \begin{Bmatrix} G40 \\ G41 \\ G42 \end{Bmatrix} \begin{Bmatrix} G00 \\ G01 \end{Bmatrix}$ X_Y_ Z_D_

说明：

G40：取消刀具半径补偿；
G41：左刀补(在刀具前进方向左侧补偿)，如图 3.3.5(a)；
G42：右刀补(在刀具前进方向右侧补偿)，如图 3.3.5(b)；
G17：刀具半径补偿平面为 *XY* 平面；
G18：刀具半径补偿平面为 *ZX* 平面；
G19：刀具半径补偿平面为 *YZ* 平面；
X, Y, Z：G00/G01 的参数，即刀补建立或取消的终点（注：投影到补偿平面上的刀具轨迹受到补偿）；
D：G41/G42 的参数，即刀补号码(D00~D99)，它代表了刀补表中对应的半径补偿值。

G40、G41、G42 都是模态代码，可相互注销。



注意：

(1) 刀具半径补偿平面的切换必须在补偿取消方式下进行；

(2) 刀具半径补偿的建立与取消只能用 G00 或 G01 指令，不得是 G02 或 G03。

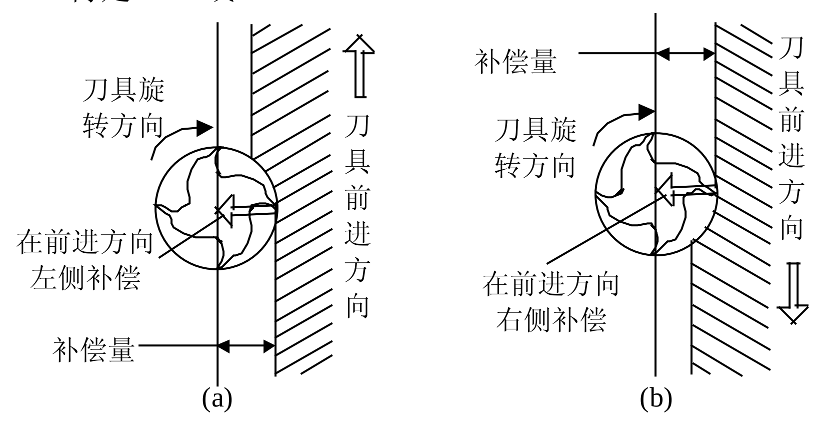
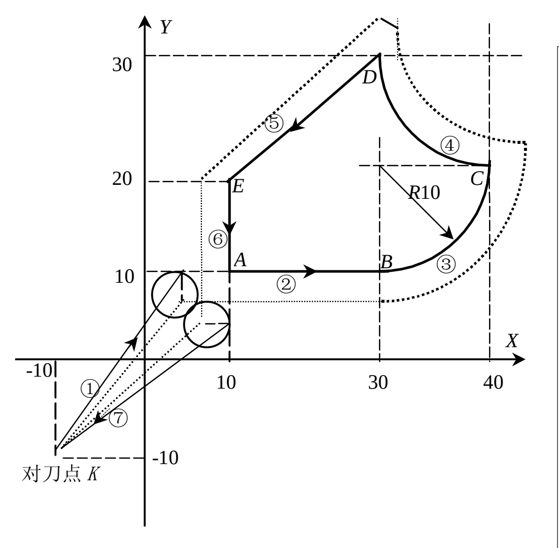


图 3.3.15 刀具补偿方向

(a)左刀补 (b)右刀补

例 11：考虑刀具半径补偿，编制图 3.3.16 所示零件的加工程序：要求建立如图所示的工件坐标系，按箭头所指示的路径进行加工，设加工开始时刀具距离工件上表面 50mm，切削深度为 10mm。



一个完整的零件程序

```
%1008
G92 X–10 Y–10 Z50
G90 G17
G42 G00 X4 Y10 D01
Z2 M03 S900
G01 Z-10 F800
X30
G03 X40 Y20 I0 J10
G02 X30 Y30 I0 J10
G01 X10 Y20
Y5
G00 Z50 M05
G40 X–10 Y–10 M02
```

图 3.3.16 刀具半径补偿编程

注意：



(1) 加工前应先用手动方式对刀，将刀具移动到相对于编程原点（–10，–10，50)的对刀点处；

(2) 图中带箭头的实线为编程轮廓，不带箭头的虚线为刀具中心的实际路线。

(2) 刀具长度补偿 G43，G44，G49

格式： $\begin{Bmatrix} G17 \\ G18 \\ G19 \end{Bmatrix} \begin{Bmatrix} G43 \\ G44 \\ G49 \end{Bmatrix} \begin{Bmatrix} G00 \\ G01 \end{Bmatrix}$ X_Y_Z_H_

说明：

G17：刀具长度补偿轴为 *Z* 轴；

G18：刀具长度补偿轴为 *Y* 轴；

G19：刀具长度补偿轴为 *X* 轴；

G49：取消刀具长度补偿；

G43：正向偏置(补偿轴终点加上偏置值)；

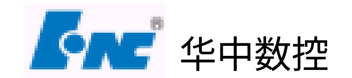
G44：负向偏置(补偿轴终点减去偏置值)；

X, Y, Z：G00/G01 的参数，即刀补建立或取消的终点；

H：G43/G44 的参数，即刀具长度补偿偏置号(H00~H99)，它代表了刀补表中对应的长度补偿值。

G43、G44、G49 都是模态代码，可相互注销。

例 12：考虑刀具长度补偿，编制如图 3.3.17 所示零件的加工程序：要求建立如图所示的工件坐标系，按箭头所指示的路径进行加工。



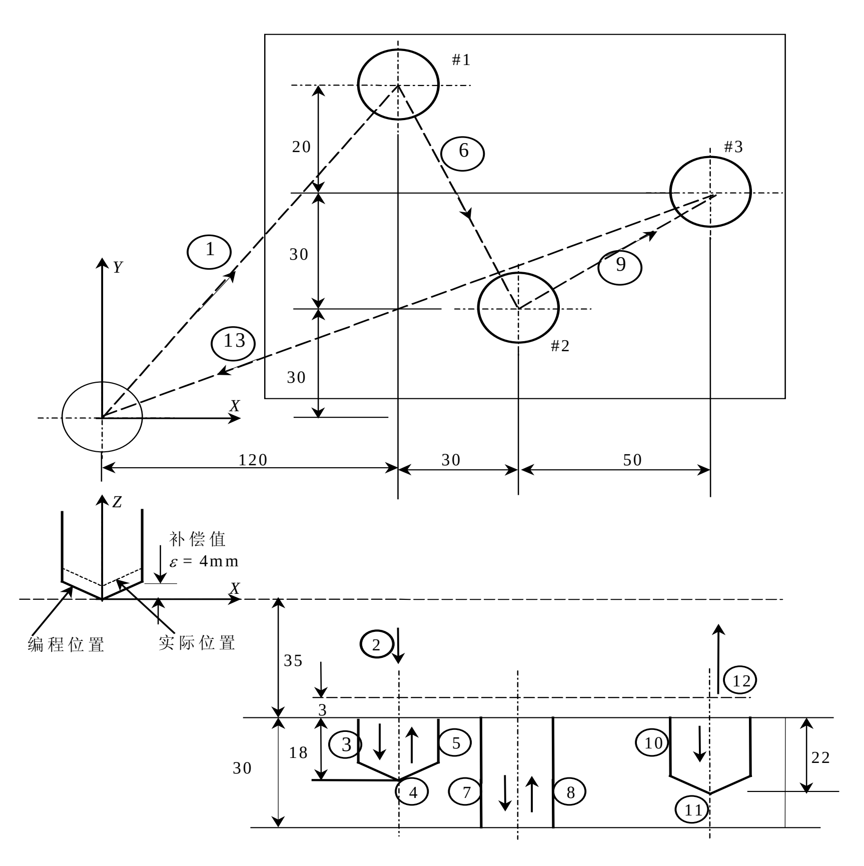
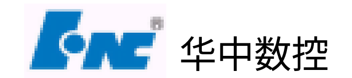


图 3.3.17 刀具长度补偿加工

```
%1050
G92 X0 Y0 Z0
G91 G00 X120 Y80 M03 S600
G43 Z–32 H01
G01 Z–21 F300
G04 P2
G00 Z21
X30 Y-50
G01 Z–41
G00 Z41
X50 Y30
G01 Z–25
G04 P2
G00 G49 Z57
X–200 Y–60 M05 M30
```



注意：

(1) 垂直于 G17/G18/G19 所选平面的轴受到长度补偿；

(2) 偏置号改变时，新的偏置值并不加到旧偏置值上，例如：

设 H01 的偏置值为 20，H02 的偏置值为 30，则

```
G90 G43 Z100 H01        ；Z 将达到 120
G90 G43 Z100 H02        ；Z 将达到 130
```

3.3.6 其他功能指令

(1) 暂停指令 G04

格式：G04　P_

说明：

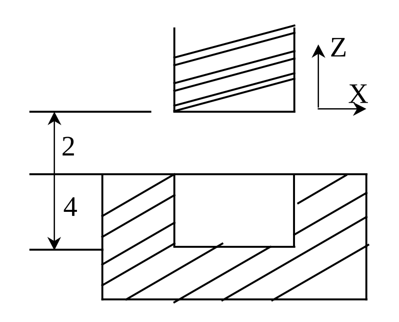
P：暂停时间，单位为 s。

G04 在前一程序段的进给速度降到零之后才开始暂停动作。

在执行含 G04 指令的程序段时，先执行暂停功能。

G04 为非模态指令，仅在其被规定的程序段中有效。

例 13：编制图 3.3.18 所示零件的钻孔加工程序。




```
%0004
G92 X0 Y0 Z0
G91 F200 M03 S500
G43 G01 Z-6 H01
G04 P5
G49 G00 Z6 M05 M30
```

图 3.3.18　G04 编程

G04 可使刀具作短暂停留，以获得圆整而光滑的表面。如对不通孔作深度控制时，在刀具进给到规定深度后，用暂停指令使刀具作非进给光整切削，然后退刀，保证孔底平整。

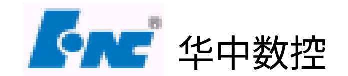
(2) 准停检验 G09

格式：G09

说明：

一个包括 G09 的程序段在继续执行下个程序段前，准确停止在本程序段的终点。该功能用于加工尖锐的棱角。

G09 为非模态指令，仅在其被规定的程序段中有效。



(2) 段间过渡方式 G61，G64

格式：$\begin{Bmatrix} G61 \\ G64 \end{Bmatrix}$

说明：

G61：精确停止检验；

G64：连续切削方式。

在 G61 后的各程序段编程轴都要准确停止在程序段的终点，然后再继续执行下一程序段。

在 G64 之后的各程序段编程轴刚开始减速时（未到达所编程的终点）就开始执行下一程序段。但在定位指令(G00，G60)或有准停校验 (G09)的程序段中，以及在不含运动指令的程序段中，进给速度仍减速到 0 才执行定位校验。

G61 方式的编程轮廓与实际轮廓相符。

G61 与 G09 的区别在于 G61 为模态指令。

G64 方式的编程轮廓与实际轮廓不同。其不同程度取决于 F 值的大小及两路径间的夹角，F 越大，其区别越大。

G61、G64 为模态指令，可相互注销，G64 为缺省值。

例 14：编制如图 3.3.19 所示轮廓的加工程序：要求编程轮廓与实际轮廓相符。

例 15：编制如图 3.3.20 所示轮廓的加工程序：要求程序段间不停顿。

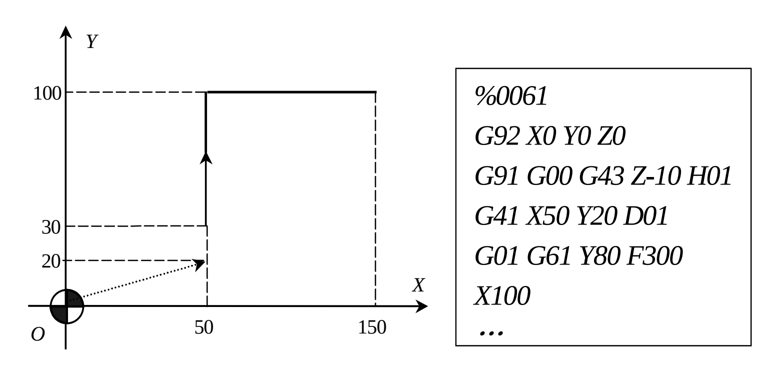
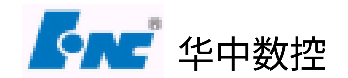


图 3.3.19　G61 编程



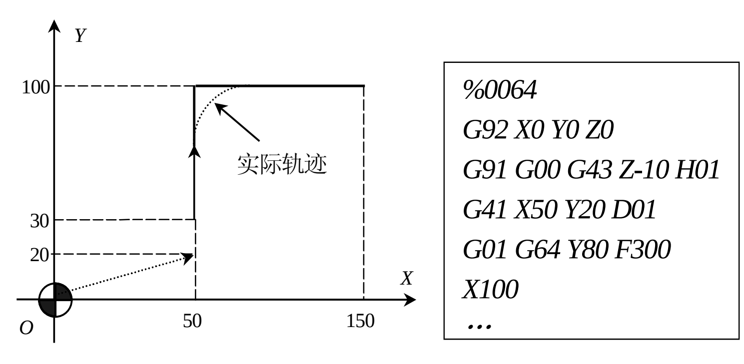


图 3.3.20　G64 编程

3.3.7 简化编程指令

(1) 镜像功能 G24，G25

格式： G24 X__Y__Z__A__

M98 P_

G25 X__Y__Z__A__

说明：

G24：建立镜像；

G25：取消镜像；

X、*Y*、*Z*、*A*：镜像位置。

当工件相对于某一轴具有对称形状时，可以利用镜像功能和子程序，只对工件的一部分进行编程，而能加工出工件的对称部分，这就是镜像功能。

当某一轴的镜像有效时，该轴执行与编程方向相反的运动。

G24、G25 为模态指令，可相互注销，G25 为缺省值。

例 16：使用镜像功能编制如图 3.3.21 所示轮廓的加工程序：设刀具起点距工件上表面 100mm，切削深度 5mm。

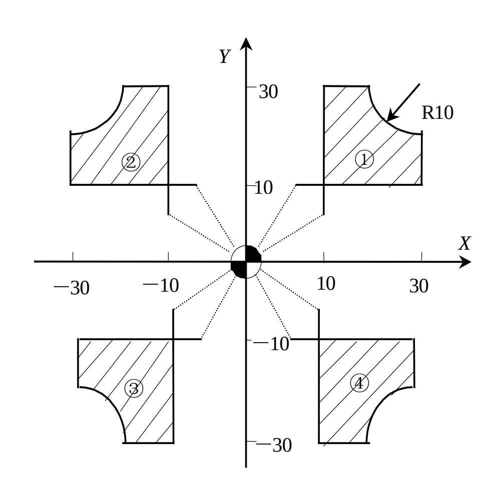
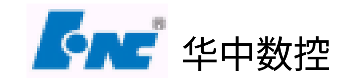


图 3.3.21 镜像功能

```
%0024                          ；主程序
      G92 X0 Y0 Z0
      G91 G17 M03 S600
      M98 P100                 ；加工①
      G24 X0                   ；Y 轴镜像，镜像位置为 X=0
      M98 P100                 ；加工②
      G24 Y0                   ；X、Y 轴镜像，镜像位置为(0，0)
      M98 P100                 ；加工③
      G25 X0                   ；X 轴镜像继续有效，取消 Y 轴镜像
      M98 P100                 ；加工④
      G25 Y0                   ；取消镜像
      M30
%100                           ；子程序(①的加工程序)：
      N100 G41 G00 X10 Y4 D01
      N120 G43 Z–98 H01
      N130 G01 Z–7 F300
      N140 Y26
      N150 X10
      N160 G03 X10 Y–10 I10 J0
      N170 G01 Y–10
      N180 X–25
      N185 G49 G00 Z105
      N200 G40 X–5 Y–10
      N210 M99
```

(2) 缩放功能 G50，G51

格式： G51 X_Y_Z_P_

M98 P_

G50

说明：

G51：建立缩放；

G50：取消缩放；

X、*Y*、*Z*：缩放中心的坐标值；

P：缩放倍数。

G51 既可指定平面缩放，也可指定空间缩放。

在 G51 后，运动指令的坐标值以（*X*，*Y*，*Z*）为缩放中心，按 *P* 规定的缩放比例进行计算。

在有刀具补偿的情况下，先进行缩放，然后才进行刀具半径补偿、刀具长度补偿。

G51、G50 为模态指令，可相互注销，G50 为缺省值。

例 17：使用缩放功能编制如图 3.3.22 所示轮廓的加工程序：已知三角形 *ABC* 的顶点为 *A*(10, 30)，*B*(90, 30)，*C*(50, 110)，三角形 *A'B'C'*是缩放后的图形，其中'缩放中心为 *D*(50, 50)，缩放系数为 0.5 倍，设刀具起点距工件上表面 50mm。

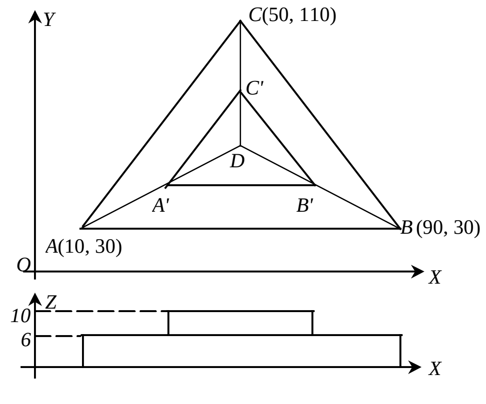
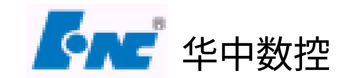


图 3.3.22 ⊿*ABC* 缩放示意图

```
%0051                          ；主程序
    G92 X0 Y0 Z60
    G91 G17 M03 S600 F300
    G43 G00 X50 Y50 Z-46 H01
    #51=14
```




```
        M98 P100              ; 加工三角形 ABC
        #51=8
        G51 X50 Y50 P0.5      ; 缩放中心(50, 50)，缩放系数 0.5
        M98 P100              ; 加工三角形 A'B'C'
        G50                   ; 取消缩放
        G49 Z46
        M05 M30
    %100                      ; 子程序(三角形 ABC 的加工程序)：
        N100 G42 G00 X-44 Y-20 D01
        N120 Z[-#51]
        N150 G01 X84
        N160 X-40 Y80
        N170 X–44 Y-88
        N180 Z[#51]
        N200 G40 G00 X44 Y28
        N210 M99
```

(3) 旋转变换 G68，G69

格式：G17 G68 X__Y__P__
G18 G68 X__Z__P__
G19 G68 Y__Z__P__
M98 P_
G69

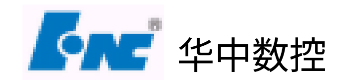
说明：

G68：建立旋转；
G69：取消旋转；
X、*Y*、*Z*：旋转中心的坐标值；
P：旋转角度，单位是(°)，0≤*P*≤360°。

在有刀具补偿的情况下，先旋转后刀补（刀具半径补偿、长度补偿)；在有缩放功能的情况下，先缩放后旋转。

G68、G69 为模态指令，可相互注销，G69 为缺省值。

例 18：使用旋转功能编制如图 3.3.23 所示轮廓的加工程序：设刀具起点距工件上表面 50mm，切削深度 5mm。



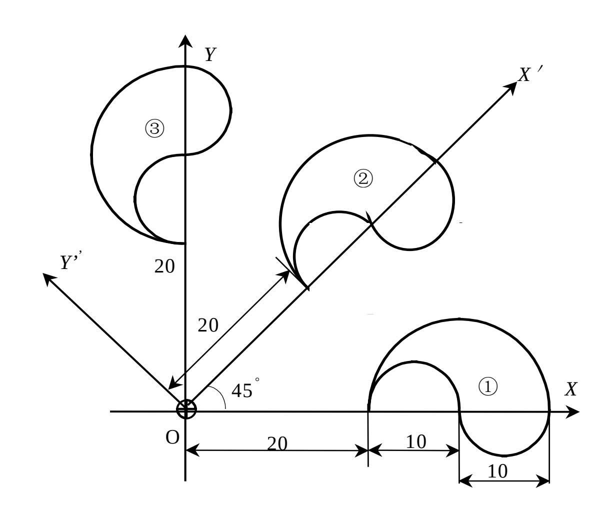


图 3.3.23　旋转变换功能

```
%0068                                   ；主程序
    N10 G92 X0 Y0 Z50
    N15 G90 G17 M03 S600
    N20 G43 Z-5 H02
    N25 M98 P200                        ；加工①
    N30 G68 X0 Y0 P45                   ；旋转 45°
    N40 M98 P200                        ；加工②
    N60 G68 X0 Y0 P90                   ；旋转 90°
    N70 M98 P200                        ；加工③
    N20 G49 Z50
    N80 G69 M05 M30                     ；取消旋转
%200                                    ；子程序 (①的加工程序)
    N100  G41 G01 X20 Y-5 D02 F300
    N105 Y0
    N110 G02 X40 I10
    N120 X30 I-5
    N130 G03 X20 I–5
    N140 G00 Y-6
    N145 G40 X0 Y0
    N150 M99
```

3.3.8 固定循环

数控加工中，某些加工动作循环已经典型化。例如，钻孔、镗孔的动作是孔位平面定位、快速引进、工作进给、快速退回等，这样一系列典型的加工动作已经预先编好程序(请参考附录 1)，存储在内存中，可用称为固定循环的一个 G 代码程序段调用，从而简化编程工作。

孔加工固定循环指令有 G73，G74，G76，G80～G89，通常由下述 6 个动作构成(见图 3.3.24)：

(1) *X*、*Y* 轴定位；
(2) 定位到 *R* 点(定位方式取决于上次是 G00 还是 G01)；
(3) 孔加工；
(4) 在孔底的动作；
(5) 退回到 *R* 点(参考点)；
(6) 快速返回到初始点。

固定循环的数据表达形式可以用绝对坐标(G90)和相对坐标(G91)表示，如图 3.3.25 所示，其中图(a)是采用 G90 的表示，图(b)是采用 G91 的表示。

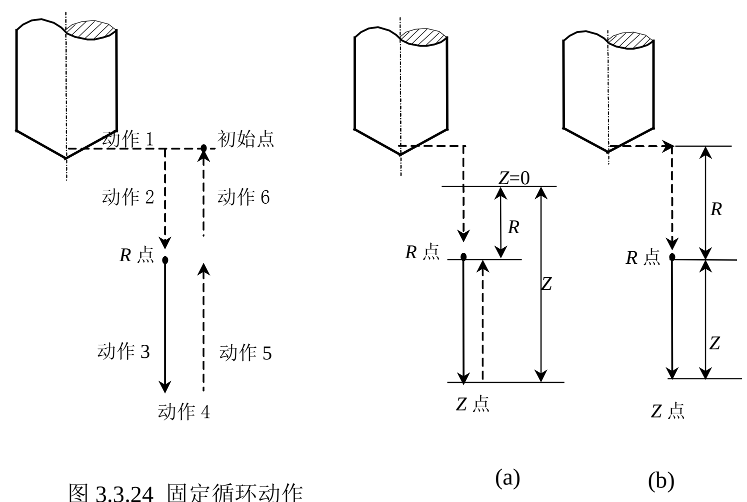


图 3.3.24 固定循环动作
实线—切削进给　虚线—快速进给

(a)　(b)
图 3.3.25 固定循环的数据形式

固定循环的程序格式包括数据形式、返回点平面、孔加工方式、孔位置数据、孔加工数据和循环次数。数据形式(G90 或 G91)在程序开始时就已指定，因此，在固定循环程序格式中可不注出。固定循环的程序格式如下：



$\begin{Bmatrix} G98 \\ G99 \end{Bmatrix}$G_X_Y_Z_R_Q_P_I_J_K_F_L_;

说明：

G98：返回初始平面；
G99：返回 *R* 点平面；
*G_*：固定循环代码 G73，G74，G76 和 G81~G89 之一；
X、*Y*：加工起点到孔位的距离(G91)或孔位坐标(G90)；
R：初始点到 *R* 点的距离(G91)或 *R* 点的坐标(G90)；
Z：*R* 点到孔底的距离(G91)或孔底坐标(G90)；
Q：每次进给深度(G73/G83)；
I、*J*：刀具在轴反向位移增量(G76/G87)；
P：刀具在孔底的暂停时间；
F：切削进给速度；
L：固定循环的次数。

G73、G74、G76 和 G81~G89、*Z*、*R*、*P*、*F*、*Q*、*I*、*J*、*K* 是模态指令。G80、G01~G03 等代码可以取消固定循环。

(1) G73：高速深孔加工循环

格式：$\begin{Bmatrix} G98 \\ G99 \end{Bmatrix}$G73 X_Y_Z_R_Q_P_K_F_L_；

说明：

Q：每次进给深度；
k：每次退刀距离。

G73 用于 *Z* 轴的间歇进给，使深孔加工时容易排屑，减少退刀量，可以进行高效率的加工。

G73 指令动作循环见图 3.3.26。

注意：Z、K、Q 移动量为零时，该指令不执行。

例 19：使用 G73 指令编制如图 3.3.26 所示深孔加工程序：设刀具起点距工件上表面 42mm，距孔底 80mm，在距工件上表面 2mm 处(R 点)由快进转换为工进，每次进给深度 10mm，每次退刀距离 5mm。

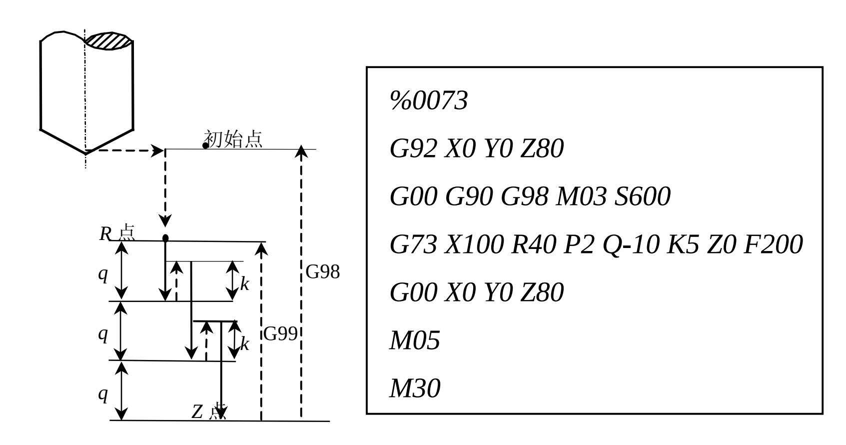


图 3.3.26　G73 指令动作图与 G73 编程

(2) G74：反攻丝循环

格式：$\begin{Bmatrix} G98 \\ G99 \end{Bmatrix}$ G74 X_Y_Z_R_P_F_L_；

G74 攻反螺纹时主轴反转，到孔底时主轴正转，然后退回。

G74 指令动作循环见图 3.3.27。

注意：

(1) 攻丝时速度倍率、进给保持均不起作用；

(2) *R* 应选在距工件表面 7mm 以上的地方；

(3) 如果 Z 的移动量为零，该指令不执行。

例 20：使用 G74 指令编制如图 3.3.27 所示反螺纹攻丝加工程序：设刀具起点距工件上表面 48mm，距孔底 60mm，在距工件上表面 8mm 处(R 点)由快进转换为工进。

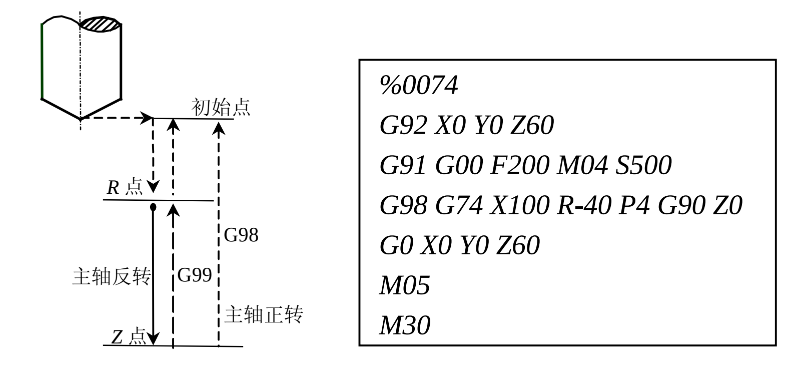
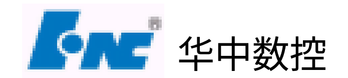


图 3.3.27　G74 指令动作图及 G74 编程



(3) G76：精镗循环

格式：$\left\{\begin{matrix}G98\\G99\end{matrix}\right\}$G76 X_Y_Z_R_P_I_J_F_L_；

说明：

I：X 轴刀尖反向位移量；

J：Y 轴刀尖反向位移量.

G76 精镗时，主轴在孔底定向停止后，向刀尖反方向移动，然后快速退刀。这种带有让刀的退刀不会划伤已加工平面，保证了镗孔精度。

G76 指令动作循环见图 3.3.28。

注意：如果 Z 的移动量为零，该指令不执行。

例 21：使用 G76 指令编制如图 3.3.28 所示精镗加工程序：设刀具起点距工件上表面 42mm，距孔底 50mm，在距工件上表面 2mm 处(R 点)由快进转换为工进。

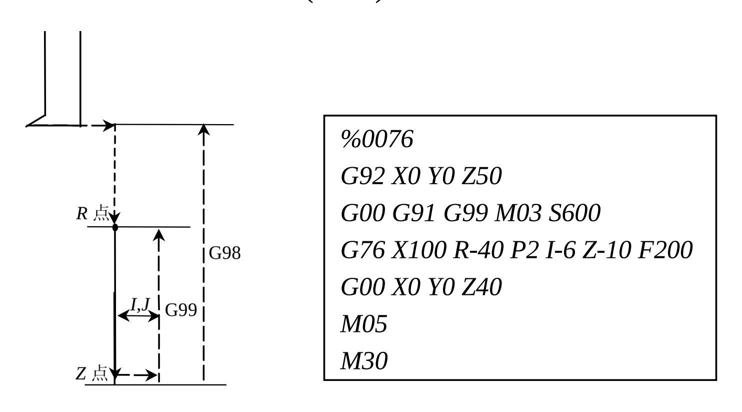


图 3.3.28 G76 指令动作图及 G76 编程

(4) G81：钻孔循环(中心钻)

格式：$\left\{\begin{matrix}G98\\G99\end{matrix}\right\}$G81 X_Y_Z_R_ F_L_；

G81 钻孔动作循环，包括 *X*，*Y* 坐标定位、快进、工进和快速返回等动作。

G81 指令动作循环见图 3.3.29。

注意：如果 Z 的移动量为零，该指令不执行。

例 22：使用 G81 指令编制如图 3.3.29 所示钻孔加工程序：设

刀具起点距工件上表面 42mm，距孔底 50mm，在距工件上表面 2mm 处(R 点)由快进转换为工进。

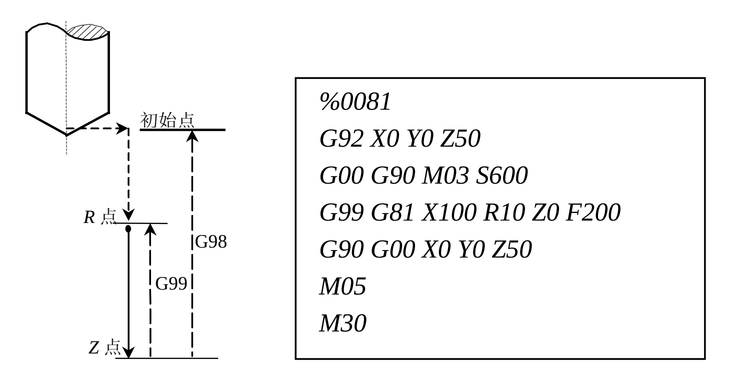


图 3.3.29　G81 指令动作图及 G81 编程

(5) G82：带停顿的钻孔循环

格式：$\begin{Bmatrix} G98 \\ G99 \end{Bmatrix}$ G82 X_Y_Z_R_P_F_L_；

G82 指令除了要在孔底暂停外，其他动作与 G81 相同。暂停时间由地址 P 给出。

G82 指令主要用于加工盲孔，以提高孔深精度。

注意：

如果 Z 的移动量为零，该指令不执行。

(6) G83：深孔加工循环

格式：$\begin{Bmatrix} G98 \\ G99 \end{Bmatrix}$ G83 X_Y_Z_R_Q_P_K_F_L_；

说明：

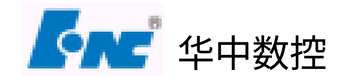
Q：每次进给深度；

k：每次退刀后，再次进给时，由快速进给转换为切削进给时距上次加工面的距离。

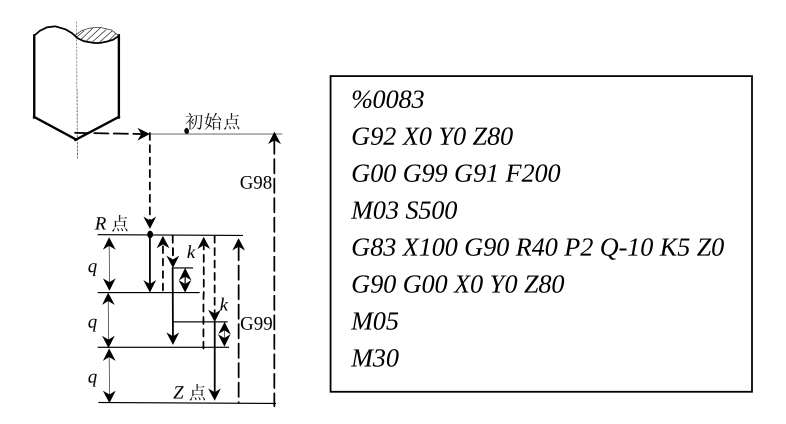
G83 指令动作循环见图 3.3.30。

注意：Z、K、Q 移动量为零时，该指令不执行。

例 23：使用 G83 指令编制如图 3.3.30 所示深孔加工程序：设刀具起点距工件上表面 42mm，距孔底 80mm，在距工件上表面 2mm 处(R 点)由快进转换为工进，每次进给深



度 10mm，每次退刀后，再由快速进给转换为切削进给时距上次加工面的距离 5mm。




```
%0083
G92 X0 Y0 Z80
G00 G99 G91 F200
M03 S500
G83 X100 G90 R40 P2 Q-10 K5 Z0
G90 G00 X0 Y0 Z80
M05
M30
```

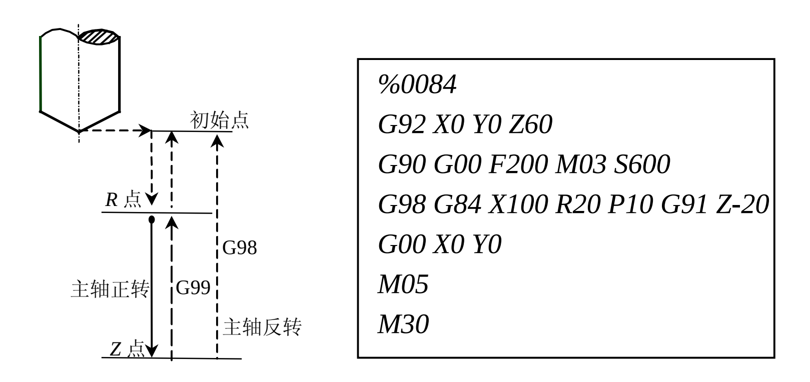
图 3.3.30　G83 指令动作图及 G83 编程

(7) G84：攻丝循环

格式：$\left\{\begin{matrix}G98\\G99\end{matrix}\right\}$ G84 X_Y_Z_R_P_F_L_；

G84 攻螺纹时从 *R* 点到 *Z* 点主轴正转，在孔底暂停后，主轴反转，然后退回。

G84 指令动作循环见图 3.3.31。




```
%0084
G92 X0 Y0 Z60
G90 G00 F200 M03 S600
G98 G84 X100 R20 P10 G91 Z-20
G00 X0 Y0
M05
M30
```

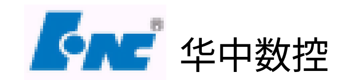
图 3.3.31　G84 指令动作图及 G84 编程

注意：

(1) 攻丝时速度倍率、进给保持均不起作用；

(2) *R* 应选在距工件表面 7mm 以上的地方；

(3) 如果 Z 的移动量为零，该指令不执行。



例 24：使用 G84 指令编制如图 3.3.31 所示螺纹攻丝加工程序：设刀具起点距工件上表面 48mm，距孔底 60mm，在距工件上表面 8mm 处(R 点)由快进转换为工进。

(8) G85：镗孔循环

G85 指令与 G84 指令相同，但在孔底时主轴不反转。

(9) G86：镗孔循环

G86 指令与 G81 相同，但在孔底时主轴停止，然后快速退回。

注意：

(1) 如果 Z 的移动位置为零，该指令不执行；

(2) 调用此指令之后，主轴将保持正转。

(10) G87：反镗循环

格式：$\begin{Bmatrix} G98 \\ G99 \end{Bmatrix}$G87 X_Y_Z_R_P_I_J_F_L_；

说明：

I：X 轴刀尖反向位移量；

J：Y 轴刀尖反向位移量.

G87 指令动作循环见图 3.3.32。描述如下

(1) 在 *X*、*Y* 轴定位；

(2) 主轴定向停止；

(3) 在 *X*、*Y* 方向分别向刀尖的反方向移动 *I*、*J* 值；

(4) 定位到 *R* 点(孔底)；

(5) 在 *X*、*Y* 方向分别向刀尖方向移动 *I*、*J* 值；

(6) 主轴正转；

(7) 在 *Z* 轴正方向上加工至 *Z* 点；

(8) 主轴定向停止；

(9) 在 *X*、*Y* 方向分别向刀尖反方向移动 *I*、*J* 值；

(10) 返回到初始点(只能用 G98)；

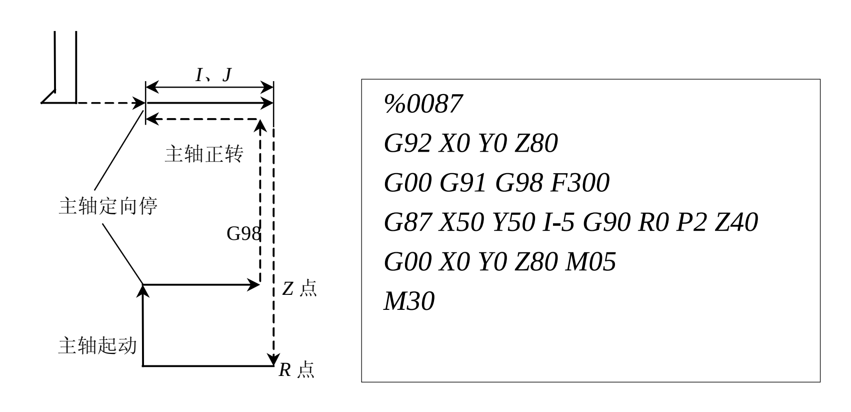
(11) 在 *X*、*Y* 方向分别向刀尖方向移动 *I*、*J* 值；

(12) 主轴正转。

注意：如果 Z 的移动量为零，该指令不执行。

例 25：使用 G87 指令编制如图 3.3.32 所示反镗加工程序：设刀具起点距工件上表面 40mm，距孔底(R 点)80mm。






```
%0087
G92 X0 Y0 Z80
G00 G91 G98 F300
G87 X50 Y50 I-5 G90 R0 P2 Z40
G00 X0 Y0 Z80 M05
M30
```

图 3.3.32　G87 指令动作图及 G87 编程

(11) G88：镗孔循环

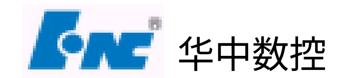
格式：$\left\{\begin{matrix}G98\\G99\end{matrix}\right\}$G88 X_Y_Z_R_P_F_L_；

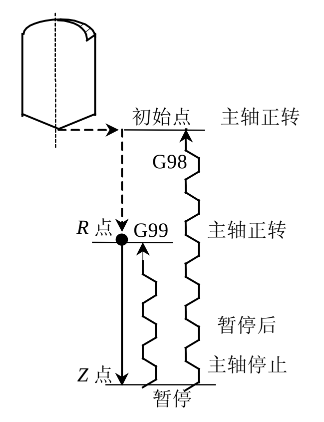
G88 指令动作循环见图 3.3.33。描述如下

(1) 在 *X*、*Y* 轴定位；
(2) 定位到 *R* 点；
(3) 在 *Z* 轴方向上加工至 *Z* 点（孔底）；
(4) 暂停后主轴停止；
(5) 转换为手动状态，手动将刀具从孔中退出；
(6) 返回到初始平面；
(7) 主轴正转。

注意：如果 Z 的移动量为零，该指令不执行。

例 26：使用 G88 指令编制如图 3.3.33 所示镗孔加工程序：设刀具起点距 R 点 40mm，距孔底 80mm。






```
%0088
G92 X0 Y0 Z80
M03 S600
G90 G00 G98 F200
G88 X60 Y80 R40 P2 Z0
G00 X0 Y0 M05
M30
```

图 3.3.33　G88 指令动作图及 G88 编程

(12) G89：镗孔循环

G89 指令与 G86 指令相同，但在孔底有暂停。

注意：如果 Z 的移动量为零，G89 指令不执行。

(13) G80：取消固定循环

该指令能取消固定循环，同时 *R* 点和 *Z* 点也被取消。

小结：

使用固定循环时应注意以下几点：

(1) 在固定循环指令前应使用 M03 或 M04 指令使主轴回转；

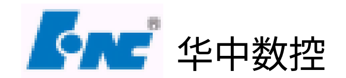
(2) 在固定循环程序段中，*X*, *Y*, *Z*, *R* 数据应至少指令一个才能进行孔加工；

(3) 在使用控制主轴回转的固定循环(G74、G84、G86)中，如果连续加工一些孔间距比较小，或者初始平面到 *R* 点平面的距离比较短的孔时，会出现在进入孔的切削动作前时，主轴还没有达到正常转速的情况，遇到这种情况时，应在各孔的加工动作之间插入 G04 指令，以获得时间；

(4) 当用 G00~G03 指令注销固定循环时，若 G00~G03 指令和固定循环出现在同一程序段，按后出现的指令运行；

(5) 在固定循环程序段中，如果指定了 M，则在最初定位时送出 M 信号，等待 M 信号完成，才能进行孔加工循环。

例 27 使用 G88 指令编制如图 3.3.34 所示的螺纹加工程序：设刀具起点距工作表面 100mm 处，切削深度为 10mm。



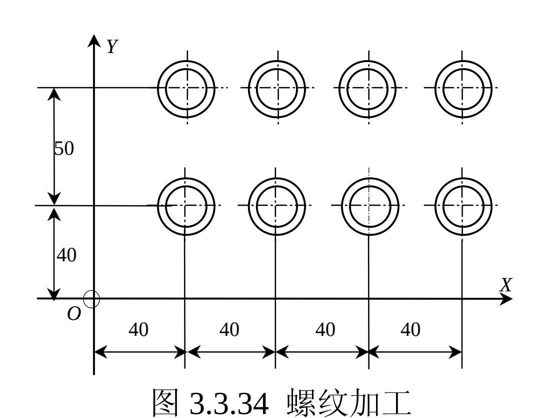


图 3.3.34 螺纹加工

(i) *先用 G81 钻孔*

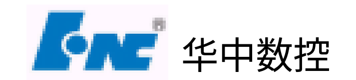
```
%1000
G92 X0 Y0 Z0
G91 G00 M03 S600
G99 G81 X40 Y40 G90 R–98 Z–110 F200
G91 X40 L3
Y50
X-40 L3
G90 G80 X0 Y0 Z0 M05
M30
```

(ii) *再用 G84 攻丝*

```
%2000
G92 X0 Y0 Z0
G91 G00 M03 S600
G99 G84 X40 Y40 G90 R–93 Z–110 F100
G91 X40 L3
Y50
X-40 L3
G90 G80 X0 Y0 Z0 M05
M30
```

3.4 宏指令编程

HNC-18iM/19iM 为用户配备了强有力的类似于高级语言的宏程序功能，用户可以使用变量进行算术运算、逻辑运算和函数的混合运算，此外宏程序还提供了循环语句、分支语句和子程序调用语句，利于编制各种复杂的零件加工程序，减少乃至免除手工



编程时进行繁琐的数值计算，以及精简程序量。

3.4.1 宏变量及常量

(1) 宏变量

#0～#49　当前局部变量
#50～#199　全局变量
#200～#249　0 层局部变量
#250～#299　1 层局部变量
#300～#349　2 层局部变量
#350～#399　3 层局部变量
#400～#449　4 层局部变量
#450～#499　5 层局部变量
#500～#549　6 层局部变量
#550～#599　7 层局部变量
#600～#699　刀具长度寄存器 H0～H99
#700～#799　刀具半径寄存器 D0～D99
#800～#899　刀具寿命寄存器

#1000 “机床当前位置X”	#1001 “机床当前位置Y”	#1002 “机床当前位置Z”
#1003 “机床当前位置A”	#1004 “机床当前位置B”	#1005 “机床当前位置C”
#1006 “机床当前位置U”	#1007 “机床当前位置V”	#1008 “机床当前位置W”
#1009 保留	#1010 “程编机床位置X”	#1011 “程编机床位置Y”
#1012 “程编机床位置Z”	#1013 “程编机床位置A”	#1014 “程编机床位置B”
#1015 “程编机床位置C”	#1016 “程编机床位置U”	#1017 “程编机床位置V”
#1018 “程编机床位置W”	#1019 保留	#1020 “程编工件位置X”
#1021 “程编工件位置Y”	#1022 “程编工件位置Z”	#1023 “程编工件位置A”
#1024 “程编工件位置B”	#1025 “程编工件位置C”	#1026 “程编工件位置U”
#1027 “程编工件位置V”	#1028 “程编工件位置W”	#1029 保留
#1030 “当前工件零点X”	#1031 “当前工件零点Y”	#1032 “当前工件零点Z”
#1033 “当前工件零点A”	#1034 “当前工件零点B”	#1035 “当前工件零点C”
#1036 “当前工件零点U”	#1037 “当前工件零点V”	#1038 “当前工件零点W”
#1039 保留	#1040 “G54 零点X”	#1041 “G54 零点Y”
#1042 “G54 零点Z”	#1043 “G54 零点A”	#1044 “G54 零点B”

#1045 “G54零点C”
#1046 “G54零点U”
#1047 “G54零点V”
#1048 “G54零点W”
#1049 保留
#1050 “G55零点X”
#1051 “G55零点Y”
#1052 “G55零点Z”
#1053 “G55零点A”
#1054 “G55零点B”
#1055 “G55零点C”
#1056 “G55零点U”
#1057 “G55零点V”
#1058 “G55零点W”
#1059 保留
#1060 “G56零点X”
#1061 “G56零点Y”
#1062 “G56零点Z”
#1063 “G56零点A”
#1064 “G56零点B”
#1065 “G56零点C”
#1066 “G56零点U”
#1067 “G56零点V”
#1068 “G56零点W”
#1069 保留
#1070 “G57零点X”
#1071 “G57零点Y”
#1072 “G57零点Z”
#1073 “G57零点A”
#1074 “G57零点B”
#1075 “G57零点C”
#1076 “G57零点U”
#1077 “G57零点V”
#1078 “G57零点W”
#1079 保留
#1080 “G58零点X”
#1081 “G58零点Y”
#1082 “G58零点Z”
#1083 “G58零点A”
#1084 “G58零点B”
#1085 “G58零点C”
#1086 “G58零点U”
#1087 “G58零点V”
#1088 “G58零点W”
#1089 保留
#1090 “G59零点X”
#1091 “G59零点Y”
#1092 “G59零点Z”
#1093 “G59零点A”
#1094 “G59零点B”
#1095 “G59零点C”
#1096 “G59零点U”
#1097 “G59零点V”
#1098 “G59零点W”
#1099 保留
#1100 “中断点位置X”
#1101 “中断点位置Y”
#1102 “中断点位置Z”
#1103 “中断点位置A”
#1104 “中断点位置B”
#1105 “中断点位置C”
#1106 “中断点位置U”
#1107 “中断点位置V”
#1108 “中断点位置W”
#1109 “坐标系建立轴”
#1110 “G28中间点位置X”
#1111 “G28中间点位置Y”
#1112 “G28中间点位置Z”
#1113 “G28中间点位置A”
#1114 “G28中间点位置B”
#1115 “G28中间点位置C”
#1116 “G28中间点位置U”
#1117 “G28中间点位置V”
#1118 “G28中间点位置W”
#1119 “G28屏蔽字”
#1120 “镜像点位置X”
#1121 “镜像点位置Y”
#1122 “镜像点位置Z”
#1123 “镜像点位置A”
#1124 “镜像点位置B”
#1125 “镜像点位置C”
#1126 “镜像点位置U”
#1127 “镜像点位置V”
#1128 “镜像点位置W”
#1129 “镜像屏蔽字”
#1130 “旋转中心(轴1)”
#1131 “旋转中心(轴2)”



#1132 “旋转角度”　#1133 “旋转轴屏蔽字”　#1134 保留

#1135 “缩放中心(轴1)”　#1136 “缩放中心(轴2)”　#1137 “缩放中心(轴3)”

#1138 “缩放比例”　#1139 “缩放轴屏蔽字”　#1140 “坐标变换代码1”

#1141 “坐标变换代码2”　#1142 “坐标变换代码3”　#1143 保留

#1144 “刀具长度补偿号”　#1145 “刀具半径补偿号”　#1146 “当前平面轴1”

#1147 “当前平面轴2”　#1148 “虚拟轴屏蔽字”　#1149 “进给速度指定”

#1150 “G代码模态值0”　#1151 “G代码模态值1”　#1152 “G代码模态值2”

#1153 “G代码模态值3”　#1154 “G代码模态值4”　#1155 “G代码模态值5

#1156 “G代码模态值6”　#1157 “G代码模态值7”　#1158 “G代码模态值8”

#1159 “G代码模态值9”　#1160 “G代码模态值10”　#1161 “G代码模态值11”

#1162 “G代码模态值12”　#1163 “G代码模态值13”　#1164 “G代码模态值14”

#1165 “G代码模态值15”　#1166 “G代码模态值16”　#1167 “G代码模态值17”

#1168 “G代码模态值18”　#1169 “G代码模态值19”　#1170 “剩余CACHE”

#1171 “备用CACHE”　#1172 “剩余缓冲区”　#1173 “备用缓冲区”

#1174 保留　#1175 保留　#1176 保留

#1177 保留　#1178 保留　#1179 保留

#1180 保留　#1181 保留　#1182 保留

#1183 保留　#1184 保留　#1185 保留

#1186 保留　#1187 保留　#1188 保留

#1189 保留　#1190 “用户自定义输入”　#1191 “用户自定义输出”

#1192 “自定义输出屏蔽”　#1193 保留　#1194 保留

(2) 常量

PI：圆周率 π

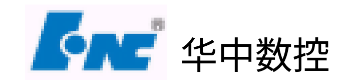
TRUE：条件成立(真)

FALSE：条件不成立(假)

3.4.2 运算符与表达式

(1) 算术运算符：

+，−，*，/



(2) 条件运算符

EQ（=），NE（≠），GT（>），
GE（≥），LT（<），LE（≤）

(3) 逻辑运算符

AND，OR，NOT

(4) 函数

SIN，COS，TAN，ATAN，ATAN2，
ABS，INT，SIGN，SQRT，EXP

(5) 表达式

用运算符连接起来的常数，宏变量构成表达式。
例如：175/SQRT[2] * COS[55 * PI/180]；
　　　　　#3*6 GT 14；

3.4.3 赋值语句

格式：宏变量=常数或表达式
把常数或表达式的值送给一个宏变量称为赋值。
例如：　#2 = 175/SQRT[2] * COS[55 * PI/180]；
　　　　　#3 = 124.0；

3.4.4 条件判别语句 IF，　ELSE，ENDIF

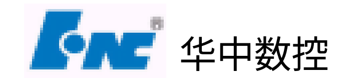
```
格式 (i)：   IF 条件表达式
             …
             ELSE
             …
             ENDIF
格式(ii) ：  IF 条件表达式
             …
             ENDIF
```

3.4.5 循环语句 WHILE，ENDW

```
格式：       WIIILE  条件表达式
                 …
                 ENDW
```

条件判别语句的使用参见宏程序编程举例。
循环语句的使用参见宏程序编程举例。

6 宏程序编制举例



例 28 切圆台与斜方台，各自加工 3 个循环，要求倾斜 10° 的斜方台与圆台相切，圆台在方台之上，顶视图见图 3.3.35。

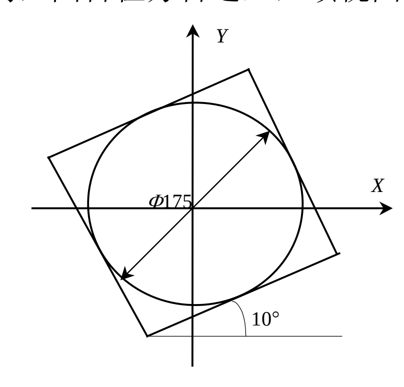


图 3.3.35 宏程序编制例图

```
%8002
#10=10.0                ；圆台阶高度
#11=10.0                ；方台阶高度
#12=124.0               ；圆外定点的 X 坐标值
#13=124.0               ；圆外定点的 Y 坐标值
N01 G92 X0.0 Y0.0 Z0.0
N05 G00 Z10.0
#0=0
N06 G00 X[–#12] Y[–#13]
N07 Z[–#10] M03 S600
WHILE   #0 LT 3         ；加工圆台
N[08+#0*6] G01 G42 X[–#12/2] Y[–175/2] F280.0 D[#0+1]
N[09+#0*6] X[0] Y[–175/2]
N[10+#0*6] G03 J[175/2]
N[11+#0*6] G01 X[#12/2] Y[–175/2]
N[12+#0*6] G40 X[#12] Y[–#13]
N[13+#0*6] G00 X[-#12] Y[–#13]
#0=#0+1
ENDW
N100 Z[-#10-#11]
#2=175/SQRT[2]*COS[55*PI/180]
#3=175/SQRT[2]*SIN[55*PI/180]
#4=175*COS[10*PI/180]
#5=175*SIN[10*PI/180]
```




```
#0=0
WHILE #0 LT 3                          ；加工斜方台
N[101+#0*6] G01 G90 G42 X[–#2] Y[–#3] F280.0 D[#0+1]
N[102+#0*6] G91 X[+#4] Y[+#5]
N[103+#0*6] X[–#5] Y[+#4]
N[104+#0*6] X[–#4] Y[–#5]
N[105+#0*6] X[+#5] Y[–#4]
N[106+#0*6] G00 G90 G40 X[–#12] Y[–#13]
#0=#0+1
ENDW
G00 X0 Y0 M05
M30
```

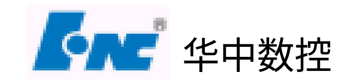


3. 附表 1 准备功能一览表

G 代码	组	功能	参数（后续地址字）	索引
G00		快速定位	X，Y，Z，4TH[注 1]	3.3.3 节
▶G01	01	直线插补	同上	
G02		顺圆插补	X，Y，Z，I，J，K，R	
G03		逆圆插补	同上	
G04	00	暂停	P	3.3.6 节
G07	16	虚轴指定	X，Y，Z，4TH	3.3.3 节
G09	00	准停校验		3.3.6 节
▶G17		XY 平面选择	X，Y	3.3.2 节
G18	02	ZX 平面选择	X，Z	
G19		YZ 平面选择	Y，Z	
G20		英寸输入		3.3.1 节
▶G21	08	毫米输入		
G22		脉冲当量		
G24	03	镜像开	X，Y，Z，4TH	3.3.7 节
▶G25		镜像关		
G28	00	返回到参考点	X，Y，Z，4TH	3.3.4 节
G29		由参考点返回	同上	
▶G40		刀具半径补偿取消		3.3.5 节
G41	09	左刀补	D	
G42		右刀补	D	
G43		刀具长度正向补偿	H	3.3.5 节
G44	10	刀具长度负向补偿	H	
▶G49		刀具长度补偿取消		
▶G50	04	缩放关		3.3.7 节
G51		缩放开	X，Y，Z，P	
G52	00	局部坐标系设定	X，Y，Z，4TH	3.3.2 节
G53		直接机床坐标系编程		
▶G54		工件坐标系 1 选择		3.3.2 节
G55		工件坐标系 2 选择		
G56	11	工件坐标系 3 选择		
G57		工件坐标系 4 选择		
G58		工件坐标系 5 选择		
G59		工件坐标系 6 选择		



G60	00	单方向定位	X，Y，Z，4TH	3.3.3 节
▶G61 G64	12	精确停止校验方式 连续方式		3.3.6 节
G65	00	子程序调用	P，A～Z	3.1.1 节
G68 ▶G69	05	旋转变换 旋转取消	X，Y，Z，P	3.3.7 节
G73 G74 G76 ▶G80 G81 G82 G83 G84 G85 G86 G87 G88 G89	06	深孔钻削循环 逆攻丝循环 精镗循环 固定循环取消 定心钻循环 钻孔循环 深孔钻循环 攻丝循环 镗孔循环 镗孔循环 反镗循环 镗孔循环 镗孔循环	X，Y，Z，P，Q， R，I，J，K 同上 同上 同上 同上 同上 同上 同上 同上 同上 同上 同上 同上	3.3.8 节
▶G90 G91	13	绝对值编程 增量值编程		3.3.2 节
G92	00	工件坐标系设定	X，Y，Z，4TH	3.3.2 节
▶G94 G95	14	每分钟进给 每转进给		3.3.1 节
▶G98 G99	15	固定循环返回起始点 固定循环返回到 R 点		3.3.8 节

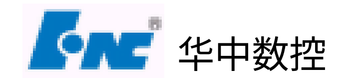


注意：

（1） 4TH 指的是 X、Y、Z 之外的第 4 轴，可用 A、B、C 等命名；

（2） 00 组中的 G 代码是非模态的，其他组的 G 代码是模态的；

（3） ▶ 标记者为缺省值。



4.　附录 1　HNC-18iM/19iM 固定循环宏程序

(1) 固定循环指令的实现及子程序调用的参数传递

HNC-18iM/19iM 的固定循环指令采用宏程序方法实现，这些宏程序调用具有模态功能。

由于各数控公司定义的固定循环含义不尽一致，采用宏程序实现固定循环，用户可按自己的要求定制固定循环，十分方便。华中数控随售出的数控装置赠送固定循环宏程序的源代码 O000。

为便于用户阅读下面固定循环宏程序的源代码，先介绍一下 HNC-18iM/19iM 宏程序/子程序调用的参数传递规则。

G 代码在调用宏(子程序或固定循环，下同)时，系统会将当前程序段各字段(A~Z 共 26 字段，如果没有定义则为零)的内容拷贝到宏执行时的局部变量#0-#25，同时拷贝调用宏时当前通道九个轴的绝对位置(机床绝对坐标)到宏执行时的局部变量#30-#38。

调用一般子程序时，不保存系统模态值，即子程序可修改系统模态并保持有效；而调用固定循环时，保存系统模态值，即固定循环子程序不修改系统模态。

下表列出了宏当前局部变量#0~#38 所对应的宏调用者传递的字段参数名。

宏当前局部变量	宏调用时所传递的字段名或系统变量
#0	A
#1	B
#2	C
#3	D
#4	E
#5	F
#6	G
#7	H
#8	I
#9	J
#10	K
#11	L
#12	M
#13	N
#14	O

#15	P
#16	Q
#17	R
#18	S
#19	T
#20	U
#21	V
#22	W
#23	X
#24	Y
#25	Z
#26	固定循环指令初始平面 Z 模态值
#27	不用
#28	不用
#29	不用
#30	调用子程序时轴 0 的绝对坐标
#31	调用子程序时轴 1 的绝对坐标
#32	调用子程序时轴 2 的绝对坐标
#33	调用子程序时轴 3 的绝对坐标
#34	调用子程序时轴 4 的绝对坐标
#35	调用子程序时轴 5 的绝对坐标
#36	调用子程序时轴 6 的绝对坐标
#37	调用子程序时轴 7 的绝对坐标
#38	调用子程序时轴 8 的绝对坐标

对于每个局部变量，都可用系统宏 AR[]来判别该变量是否被定义，是被定义为增量或绝对方式。该系统宏的调用格式如下：

AR[#变量号]

返回：

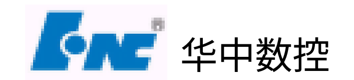
0：　表示该变量没有被定义；

90：表示该变量被定义为绝对方式 G90；

91：表示该变量被定义为相对方式 G91。

例：下面的主程序 O1000 在调用子程序 O9990 时，设置了 I、J、K 之值，子程序 O9990 可分别通过当前局部变量#8、#9、#10 来访问主程序的 I、J、K 之值。

```
%1000
```




```
G92 X0Y0Z0
M98 P9990 I20 J30 K40
M30
%9990
IF [AR[#8] EQ 0] OR [AR[#9] EQ 0] OR [AR[#10] EQ 0]
                ; 如果没有定义 I、J、K 值，
    M99         ; 则返回
ENDIF
N10 G91                    ; 用增量方式编写宏程序
IF AR[#8] EQ 90 ; 如果 I 值是绝对方式 G90
    #8=#8-#30      ;将 I 值转换为增量方式, #30 为 X 的绝对坐标
ENDIF
…
M99
```

HNC-18iM/19iM 子程序嵌套调用的深度最多可以有九层，每一层子程序都有自己独立的局部变量（变量个数为 50）。当前局部变量为#0-#49，第一层局部变量为#200-#249，第二层局部变量为#250-#299，第三层局部变量#300-#349，依此类推。

在子程序中如何确定上层的局部变量，要依上层的层数而定，例：

```
o0099
g92 X0 Y0 Z0
N100 #10=98
M98 P100
M30

o100
N200 #10=100     ;此时 N100 所在段的局部变量#10 为第一层#210
M98 P110
M99

o110
N300 #10=200     ;此时 N200 所在段的局部变量为第二层#260
                 ; N100 所在段的局部变量#10 为第一层#210
M99
```

为了更深入地了解 HNC-18iM/19iM 宏程序，这里给出一个利用小



直线段逼近整园的数控加工程序：

```
O1000
G92 X0 Y0 Z0
M98 P2 X-50 Y0 R50                ;宏程序调用，加工整圆
M30

O2
; 加工整圆子程序，园心为(X,Y),半径为 R
; X -> #23     Y -> #24       R -> #17
IF [AR[#17] EQ 0] OR [#17 EQ 0]   ;如果没有定义 R
    M99
ENDIF

IF [ AR[#23] EQ 0 ] OR [ AR[#24] EQ 0 ] ;如果没有定义圆心
    M99
ENDIF

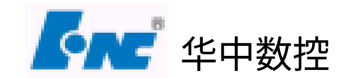
#45=#1162      ; 记录第 12 组模态码#1162，是 G61 或 G64?
#46=#1163      ; 记录第 13 组模态码#1163，是 G90 OR G91?

G91 G64                 ;用相对编程 G91 及连续插补方式 G64
IF [ AR[#23] EQ 90 ]  ;如果 X 为绝对编程方式
    #23=#23-#30           ;    则转为相对编程方式
ENDIF

IF [ AR[#24] EQ 90 ]  ;  如果 Y 为绝对编程方式
    #24 = #24-#31     ;     则转为相对编程方式
ENDIF

#0=#23+#17*COS[0];
#1=#24+#17*SIN[0];
G01 X[#0] Y[#1];

#10=1
WHILE [#10 LE 100]        ;用 100 段小直线逼近圆
    #0 = #17*[ COS[#10*2*PI/100]-COS[[#10-1]*2*PI/100] ]
```




```
        #1 = #17*[ SIN[#10*2*PI/100]-SIN[[#10-1]*2*PI/100] ]
        G01 X[#0] Y[#1]
        #10=#10+1
    ENDW

    G[#45] G[#46]          ;恢复第 12 组、13 组模态
    M99
```

(2) 固定循环指令的宏程序实现

HNC-18iM/19iM 的固定循环宏程序源代码请参考文件 O000，下面是该文件的内容。

```
%0000
；G73,G74,G76,G80,G81,G82，G83,G84,G85,G86,G87,G88,G89 宏程序定义
；_______________________________________________________________
%0073
；G73 宏程序实现源代码，调用本程序之前，必须转动主轴 M03 或 M04

IF [AR[#25] EQ 0] OR [AR[#16] EQ 0] OR [AR[#10] EQ 0]
    M99；如果没有定义孔底 Z 值、每次进给深度 Q 值或退刀量 K，则返回
ENDIF

N10 G91          ；用增量方式编写宏程序

IF AR[#23] EQ 90     ；如果 X 值是绝对方式 G90
    #23=#23-#30 ；将 X 转换为增量，#30 为调用本程序时 X 的绝对坐标
ENDIF

IF AR[#24] EQ 90     ；如果 Y 值是绝对方式 G90
    #24=#24-#31 ；将 Y 转换为增量，#31 为调用本程序时 Y 的绝对坐标
ENDIF

IF AR[#17] EQ 90     ；如果参考点平面 R 值是绝对方式 G90
    #17=#17-#32 ；将 R 转换为增量，#32 为调用本程序时 Z 的绝对坐标
ELSE
    IF AR[#26] NE 0 ；初始 Z 平面模态值存在
        #17=#17+#26-#32 ；则将 R 值转换为增量方式
```




```
    ENDIF
ENDIF

IF AR[#25] EQ 90     ; 如果孔底 Z 值是绝对方式 G90
    #25=#25-#32-#17 ; 将 Z 值转换为增量
ENDIF

IF [#25 GE 0] OR [#16 GE 0] OR [#10 LE 0] OR [#10 GE [-#16]]
        ; 如果增量方式的 Z、Q>=0 或退刀量 K<=0 或 K>Q 的绝对值
    M99 ; 则返回
ENDIF

N20 X[#23] Y[#24]    ; 移到 XY 孔加工位
N30 Z[#17]           ; 移到参考点 R

#40=-#25                 ; 循环变量#40 初始值为参考点到孔底的位移
量
#41=0                ; 循环变量#41 为退刀量
WHILE #40 GT [-#16] ; 如果还可以进刀一次
    N50 G01 Z[#16-#41]  ; 进刀
    N55 G04 P0.1            ; 暂停
    N60 G00 Z[#10]      ; 退刀
    N65 G04 P0.1            ; 暂停
    #41=#10             ; 退刀量
    #40=#40+#16         ; 进刀量为负数，#40 将减少
ENDW

N70 G01 Z[-#40-#41]      ; 最后一刀到孔底
N80 G04 P[#15]           ; 在孔底暂停

IF #1165 EQ 99           ;如果第 15 组 G 代码模态值为 G99，
    N90 G00 Z[-#25]      ;即返回参考点 R 平面
ELSE                         ;否则
    IF AR[#26] EQ 0
        N90 G00 Z[-#25-#17] ;返回初始平面, 注：#25 及#17 均为负数
    ELSE
```




```
        N90 G90 G00 Z[#26]  ;否则返回初始平面
    ENDIF
ENDIF

M99

;_______________________________________________________________
%0074
;反攻丝循环 G74 的宏程序实现源代码，调用本程序后，主轴反转

IF AR[#25] EQ 0       ; 没有定义孔底 Z 坐标
    M99
ENDIF

N10 G91               ;用增量方式编写宏程序

IF AR[#23] EQ 90          ; 如果 X 值是绝对方式 G90
    #23=#23-#30       ; 将 X 值转换为增量方式, #30 为 X 的绝对坐标
ENDIF

IF AR[#24] EQ 90          ; 如果 Y 值是绝对方式 G90
    #24=#24-#31       ; 将 Y 值转换为增量方式, #31 为 Y 的绝对坐标
ENDIF

IF AR[#17] EQ 90          ; 如果参考点 R 值是绝对方式 G90
    #17=#17-#32       ; 将 R 值转换为增量方式, #32 为 Z 的绝对坐标
ELSE
    IF AR[#26] NE 0 ; 初始 Z 平面模态值存在
        #17=#17+#26-#32 ; 将 R 值转换为增量, #32 为 Z 的绝对坐标
    ENDIF
ENDIF

IF AR[#25] EQ 90          ; 如果孔底 Z 值是绝对方式 G90
    #25=#25-#32-#17 ; 将 Z 值转换为增量方式, #32 为 Z 的绝对坐标
ENDIF
```

```
IF #25 GE 0             ; 如果增量方式的 Z 大于等于零
    M99                 ; 则返回
ENDIF

N20 X[#23] Y[#24] M04    ; 移到孔加工位 XY，并且主轴反转
N30 Z[#17]               ; 移到参考点 R
N50 G34 Z[#25]           ; 反攻丝到孔底，攻丝时进给保持将不起作用
N60 G04 P[#15]           ; 暂停
N70 M03                  ; 主轴正转

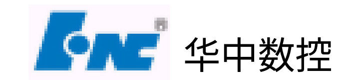
IF #1165 EQ 99           ; 如果为 G99，即返回参考点 R 平面
    N90 G34 Z[-#25]      ; 正向攻丝，攻丝时进给保持将不起作用
ELSE
    N90 G34 Z[-#25]      ; 正向攻丝，攻丝时进给保持将不起作用
    IF AR[#26] EQ 0
        N95 G00 Z[-#17] ;返回初始平面, 注：#25 及#17 均为负数
    ELSE
        N95 G90 G00 Z[#26]  ;否则返回初始平面
    ENDIF
ENDIF

N100 M04                 ; 主轴反转
M99

; ________________________________________________________________
%0076
; 精镗循环 G76 宏程序源代码，
; 调用本程序前，必须调用 M03 或 M04 指令让主轴转动。

IF [AR[#25] EQ 0] OR [[AR[#8] OR AR[#9]] EQ 0]
        ; 如果没有定义孔底 Z 值、I 或 J 退刀量，则返回
    M99 ; 则返回
ENDIF

N10 G91                 ;用增量方式编写宏程序
```




```
IF AR[#23] EQ 90          ; 如果X值是绝对方式G90
    #23=#23-#30       ; 将X值转换为增量方式，#30为X的绝对坐标
ENDIF

IF AR[#24] EQ 90          ; 如果Y值是绝对方式G90
    #24=#24-#31       ; 将X值转换为增量方式，#31为Y的绝对坐标
ENDIF

IF AR[#17] EQ 90          ; 如果参考点R值是绝对方式G90
    #17=#17-#32       ; 将R值转换为增量方式，#32为Z的绝对坐标
ELSE
    IF AR[#26] NE 0 ; 初始Z平面模态值存在
        #17=#17+#26-#32 ;将R值转换为增量，#32为Z的绝对坐标
    ENDIF
ENDIF

IF AR[#25] EQ 90          ; 如果孔底Z值是绝对方式G90
    #25=#25-#32-#17 ;将Z值转换为增量方式，#32为Z的绝对坐标
ENDIF

IF #25 GE 0           ; 如果增量方式的Z大于等于零
    M99
ENDIF

N20 X[#23] Y[#24]     ; 移到孔加工位XY,
N30 Z[#17]            ; 移到参考点R

N50 G01 Z[#25]        ; 镗孔，在此之前，必须让主轴转动
N55 M05               ; 主轴停
N60 M19               ; 主轴定向
N70 G04 P[#15]        ; 暂停
N80 G00 X[#8] Y[#9] ; 让刀

IF #1165 EQ 99        ; 如果第15组G代码模态值为G99
    N90 G00 Z[-#25]       ; 即返回参考点R平面
ELSE
```




```
    IF AR[#26] EQ 0
        N90 G00 Z[-#25-#17] ;否则返回初始平面
    ELSE
        N90 G90 G00 Z[#26]  ;否则返回初始平面
    ENDIF
ENDIF
M99

;______________________________________________________________
%0081
; 钻孔循环(中心钻)G81 宏程序实现源代码
;调用本程序之前，必须让主轴转动起来

IF AR[#25] EQ 0         ; 如果没有指定 Z 值
    M99                 ; 则返回
ENDIF

N10 G91                 ; 宏程序用增量编程 G91!

IF AR[#23] EQ 90            ; 如果 X 值是绝对编程 G90
    #23=#23-#30         ; 则改为相对编程 G91
ENDIF

IF AR[#24] EQ 90            ; 如果 Y 值是绝对编程 G90
    #24=#24-#31         ; 则改为相对编程 G91
ENDIF

IF AR[#17] EQ 90            ; 如果 R 值是绝对编程 G90
    #17=#17-#32         ; 则改为相对编程 G91
ELSE
    IF AR[#26] NE 0         ; 初始 Z 平面模态值存在
        #17=#17+#26-#32 ; 则将 R 值转换为增量方式
    ENDIF
ENDIF

IF AR[#25] EQ 90            ; 如果 Z 值是绝对编程 G90
```




```
    #25=#25-#32-#17 ; 则改为相对编程 G91
ENDIF

IF #25 GE 0           ; 如果 Z 值相对当前点不下降
    M99               ; 则返回
ENDIF

N20 X[#23] Y[#24]     ; 移到 XY 起始点
N30 Z[#17]            ; 移到参考点 R
N50 G01 Z[#25]        ; 钻孔到孔底 Z 点

IF #1165 EQ 99        ; 如果第 15 组 G 代码模态值为 G99
    N90 G00 Z[-#25] ; 则返回参考点 R 平面
ELSE
    IF AR[#26] EQ 0
        N90 G00 Z[-#25-#17] ; 否则返回初始平面
    ELSE
        N90 G90 G00 Z[#26]  ;否则返回初始平面
    ENDIF
ENDIF

M99


;________________________________________________________________
%0082
; 带停顿的钻孔循环(中心钻)G82 宏程序实现源代码
;调用本程序之前，必须让主轴转动起来

IF AR[#25] EQ 0       ; 如果没有指定 Z 值
    M99               ; 则返回
ENDIF

N10 G91               ;宏程序用增量编程 G91!

IF AR[#23] EQ 90          ; 如果 X 值是绝对编程 G90
```




```
    #23=#23-#30        ; 则改为相对编程 G91
ENDIF

IF AR[#24] EQ 90           ; 如果 Y 值是绝对编程 G90
    #24=#24-#31        ; 则改为相对编程 G91
ENDIF

IF AR[#17] EQ 90           ; 如果 R 值是绝对编程 G90
    #17=#17-#32        ; 则改为相对编程 G91
ELSE
    IF AR[#26] NE 0 ; 初始 Z 平面模态值存在
        #17=#17+#26-#32 ; 则将 R 值转换为增量方式
    ENDIF
ENDIF

IF AR[#25] EQ 90           ; 如果 Z 值是绝对编程 G90
    #25=#25-#32-#17 ; 则改为相对编程 G91
ENDIF

IF #25 GE 0                ; 如果 Z 值相对当前点不下降
    M99                ; 则返回
ENDIF

N20 X[#23] Y[#24]      ; 移到 XY 起始点
N30 Z[#17]             ; 移到参考点 R
N50 G01 Z[#25]         ; 钻孔到孔底 Z 点
N60 G04 P[#15]         ; 在孔底暂停

IF #1165 EQ 99         ; 如果第 15 组 G 代码模态值为 G99
    N90 G00 Z[-#25] ; 则返回参考点 R 平面
ELSE
    IF AR[#26] EQ 0
        N90 G00 Z[-#25-#17] ;否则返回初始平面
    ELSE
        N90 G90 G00 Z[#26]  ;否则返回初始平面
    ENDIF
```




```
ENDIF

M99
;_____________________________________________________________
%0083
; 深孔加工循环 G83 宏程序实现源代码,调用本程序前必须让主轴转起来

IF [AR[#25] EQ 0] OR [AR[#16] EQ 0] OR [AR[#10] EQ 0]
        ; 如果没有定义孔底 Z 值、每次进给深度 Q 值或退刀量 K
    M99 ; 则返回
ENDIF

N10 G91             ; 用增量方式编写宏程序

IF AR[#23] EQ 90        ; 如果 X 值是绝对方式 G90
    #23=#23-#30     ; 则将 X 值转换为增量方式
ENDIF

IF AR[#24] EQ 90        ; 如果 Y 值是绝对方式 G90
    #24=#24-#31     ; 则将 Y 值转换为增量方式
ENDIF

IF AR[#17] EQ 90        ; 如果参考点平面 R 值是绝对方式 G90
    #17=#17-#32     ; 则将 R 值转换为增量方式
ELSE
    IF AR[#26] NE 0 ; 初始 Z 平面模态值存在
        #17=#17+#26-#32 ; 则将 R 值转换为增量方式
    ENDIF
ENDIF

IF AR[#25] EQ 90        ; 如果孔底 Z 值是绝对方式 G90
    #25=#25-#32-#17 ; 则将 Z 值转换为增量方式
ENDIF
IF [#25 GE 0] OR [#16 GE 0] OR [#10 LE 0] OR [#10 GE [-#16]]
        ; 如果增量方式的 Z、Q>=0 或退刀量 K<=0 或 K>Q 的绝对值
    M99 ; 则返回
```




```
ENDIF

N20 X[#23] Y[#24]    ; 移到 XY 起始点
N30 Z[#17]           ; 移到参考点 R

#40=-#25
#41=0
#42=0
WHILE #40 GT [-#16]          ; 如果还可以进刀一次
    N40 G01 Z[#16-#42]       ; 进刀
    N50 G04 P0.1                 ; 暂停
    N55 G00 Z[-#16-#41]      ; 退刀
    N60     Z[#16+#41+#10]   ; 快速回到已加工面
    N65 G04 P0.1                 ; 暂停
    #42=#10
    #41=#41+#16
    #40=#40+#16
ENDW

N70 G01 Z[-#40-#42]          ; 最后一次进刀
N80 G04 P[#15]               ; 暂停

IF #1165 EQ 99               ; 如果第 15 组 G 代码模态值为 G99
    N90 G00 Z[-#25]              ; 即返回参考点 R 平面
ELSE
    IF AR[#26] EQ 0
        N90 G00 Z[-#25-#17]  ; 否则返回初始平面
    ELSE
        N90 G90 G00 Z[#26]   ;否则返回初始平面
    ENDIF
ENDIF

M99
;________________________________________________________________
%0084
; 攻丝循环 G84 宏程序实现源代码。调用本程序之后，主轴将保持正转
```

```
IF AR[#25] EQ 0        ; 没有定义孔底 Z 平面
    M99                ; 则返回
ENDIF

N10 G91                ;用增量方式编写宏程序

IF AR[#23] EQ 90           ; 如果 X 值是绝对方式 G90
    #23=#23-#30        ; 则将 X 值转换为增量方式
ENDIF

IF AR[#24] EQ 90           ; 如果 Y 值是绝对方式 G90
    #24=#24-#31        ; 则将 Y 值转换为增量方式
ENDIF

IF AR[#17] EQ 90           ; 如果参考点平面 R 值是绝对方式 G90
    #17=#17-#32        ; 则将 R 值转换为增量方式
ELSE
    IF AR[#26] NE 0        ; 初始 Z 平面模态值存在
        #17=#17+#26-#32 ; 则将 R 值转换为增量方式
    ENDIF
ENDIF

IF AR[#25] EQ 90           ; 如果孔底 Z 值是绝对方式 G90
    #25=#25-#32-#17 ; 则将 Z 值转换为增量方式
ENDIF

IF #25 GE 0            ; 如果 Z 值相对当前点不下降
    M99                ; 则返回
ENDIF

N20 X[#23] Y[#24] M03   ; 移到定位点 XY，且主轴正转
N30 Z[#17]              ; 移到参考点 R
N50 G34 Z[#25]          ; 正向攻丝，进给保持和进给修调将不起作用
N60 G04 P[#15]          ; 在孔底暂停
N70 M04                 ; 主轴返转
```

```
IF #1165 EQ 99              ; 如果为 G99，则返回参考点 R 平面
    N90 G34 Z[-#25]         ; 反向攻丝，回到 R 点
ELSE
    N90 G34 Z[-#25]             ; 先反向攻丝，回到 R 点
    IF AR[#26] EQ 0
        N95 G00 Z[-#17]     ;再返回初始平面
    ELSE
        N90 G90 G00 Z[#26]  ;返回初始平面
    ENDIF
ENDIF

N100 M03                        ; 主轴正转
M99
;______________________________________________________________
%0085
; 镗孔循环 G85 宏程序实现源代码。调用本程序前必须让主轴转动

IF AR[#25] EQ 0         ; 没有定义孔底 Z 平面
    M99                 ; 则返回
ENDIF

N10 G91                 ;用增量方式编写宏程序

IF AR[#23] EQ 90            ; 如果 X 值是绝对方式 G90
    #23=#23-#30         ; 则将 X 值转换为增量方式
ENDIF

IF AR[#24] EQ 90            ; 如果 Y 值是绝对方式 G90
    #24=#24-#31         ; 则将 Y 值转换为增量方式
ENDIF

IF AR[#17] EQ 90            ; 如果参考点平面 R 值是绝对方式 G90
    #17=#17-#32         ; 则将 R 值转换为增量方式
ELSE
    IF AR[#26] NE 0         ; 初始 Z 平面模态值存在
```




```
        #17=#17+#26-#32 ; 则将 R 值转换为增量方式
    ENDIF
ENDIF

IF AR[#25] EQ 90        ; 如果孔底 Z 值是绝对方式 G90
    #25=#25-#32-#17 ; 则将 Z 值转换为增量方式
ENDIF

IF #25 GE 0             ; 如果 Z 值相对当前点不下降
    M99                 ; 则返回
ENDIF

N20 X[#23] Y[#24]       ; 移到定位点 XY
N30 Z[#17]              ; 移到参考点 R

N50 G01 Z[#25]          ; 孔加工
N60 G04 P[#15]          ; 在孔底暂停

IF #1165 EQ 99              ; 如果第 15 组 G 代码模态值为 G99
    N90 G01 Z[-#25]             ; 则返回参考点 R 平面
ELSE
    N90 G01 Z[-#25]             ; 否则先回到初始点
    IF AR[#26] EQ 0
        N95 g00 z[-#17]         ; 再返回初始平面
    ELSE
        N90 G90 G00 Z[#26]  ; 返回初始平面
    ENDIF
ENDIF

M99
;________________________________________________________________
%0086
; 镗孔循环 G86 宏程序实现源代码。调用本程序之后，主轴将保持正转
IF AR[#25] EQ 0         ; 没有定义孔底 Z 平面
    M99                 ; 则返回
ENDIF
```




```
N10 G91                ;用增量方式编写宏程序

IF AR[#23] EQ 90           ; 如果 X 值是绝对方式 G90
    #23=#23-#30        ; 则将 X 值转换为增量方式
ENDIF

IF AR[#24] EQ 90           ; 如果 Y 值是绝对方式 G90
    #24=#24-#31        ; 则将 Y 值转换为增量方式
ENDIF

IF AR[#17] EQ 90           ; 如果参考点平面 R 值是绝对方式 G90
    #17=#17-#32        ; 则将 R 值转换为增量方式
ELSE
    IF AR[#26] NE 0        ; 初始 Z 平面模态值存在
        #17=#17+#26-#32 ; 则将 R 值转换为增量方式
    ENDIF
ENDIF

IF AR[#25] EQ 90           ; 如果孔底 Z 值是绝对方式 G90
    #25=#25-#32-#17 ; 则将 Z 值转换为增量方式
ENDIF

IF #25 GE 0            ; 如果 Z 值相对当前点不下降
    M99                ; 则返回
ENDIF

N20 X[#23] Y[#24] M03      ; 移到孔位 XY
N30 Z[#17]                 ; 移到参考点 R

N50 G01 Z[#25]             ; 孔加工
N60 M05                    ; 主轴停

IF #1165 EQ 99             ; 如果第 15 组 G 代码模态值为 G99
    N90 G00 Z[-#25]            ; 则快速返回参考点 R 平面
ELSE
```





```
    IF AR[#26] EQ 0
        N90 G00 Z[-#25-#17] ;否则返回初始平面
    ELSE
        N90 G90 G00 Z[#26]  ;否则返回初始平面
    ENDIF
ENDIF

N95 M03                    ; 主轴正转
M99
;_______________________________________________________________
%0087
; 反镗孔循环 G87 宏程序实现源代码
; 调用本程序之后，主轴将保持正转(G99)或停(G98)

IF [AR[#25] EQ 0] OR [[AR[#8] OR AR[#9]] EQ 0]
        ; 如果没有定义孔底 Z 值、I 或 J 退刀量，则返回
    M99 ; 则返回
ENDIF

N10 G91                ;用增量方式编写宏程序

IF AR[#23] EQ 90           ; 如果 X 值是绝对方式 G90
    #23=#23-#30        ; 则将 X 值转换为增量方式
ENDIF

IF AR[#24] EQ 90           ; 如果 Y 值是绝对方式 G90
    #24=#24-#31        ; 则将 Y 值转换为增量方式
ENDIF

IF AR[#17] EQ 90           ; 如果参考点平面 R 值是绝对方式 G90
    #17=#17-#32        ; 则将 R 值转换为增量方式
ELSE
    IF AR[#26] NE 0 ; 初始 Z 平面模态值存在
        #17=#17+#26-#32 ; 则将 R 值转换为增量方式
    ENDIF
ENDIF
```




```
IF AR[#25] EQ 90        ; 如果孔底 Z 值是绝对方式 G90
    #25=#25-#32-#17 ; 则将 Z 值转换为增量方式
ENDIF

IF #25 LE 0            ; 如果 Z 值相对当前点不上升
  M99                  ; 返回
ENDIF

N20 X[#23] Y[#24]      ; 移到孔位 XY
n22 m05                ; 主轴停
N25 M19                ; 主轴定向停止
N30 X[#8] Y[#9]        ; 朝刀尖反向移动
N40 Z[#17]             ; 定位到加工面
n45 g04 p[#15]         ; 暂停

N50 G01 X[-#8] Y[-#9] M03    ; 朝刀尖正向移动，且主轴正转
N55 Z[#25]                   ; 加工
n58 m05                      ; 主轴停
N60 M19                      ; 主轴定向停止

N80 X[#8] Y[#9]              ; 朝刀尖反向移动

IF #1165 EQ 98               ; 如果要求返回初始面 G98
    IF AR[#26] EQ 0
        N90 G00 Z[-#25-#17] ; 返回初始平面
    ELSE
        N90 G90 G00 Z[#26]  ; 返回初始平面
    ENDIF
ELSE
    N100 X[-#8] Y[-#9] M03  ; 返回 R 点，主轴正转
ENDIF
M99
;______________________________________________________________
%0088
; 镗孔循环 G88 宏程序实现源代码。调用本程序之后，主轴将保持正转
```




```
IF AR[#25] EQ 0         ; 没有定义孔底 Z 平面
    M99                 ; 则返回
ENDIF

N10 G91                 ;用增量方式编写宏程序

IF AR[#23] EQ 90            ; 如果 X 值是绝对方式 G90
    #23=#23-#30         ; 则将 X 值转换为增量方式
ENDIF

IF AR[#24] EQ 90            ; 如果 Y 值是绝对方式 G90
    #24=#24-#31         ; 则将 Y 值转换为增量方式
ENDIF

IF AR[#17] EQ 90            ; 如果参考点平面 R 值是绝对方式 G90
    #17=#17-#32         ; 则将 R 值转换为增量方式
ELSE
    IF AR[#26] NE 0         ; 初始 Z 平面模态值存在
        #17=#17+#26-#32 ; 则将 R 值转换为增量方式
    ENDIF
ENDIF

IF AR[#25] EQ 90            ; 如果孔底 Z 值是绝对方式 G90
    #25=#25-#32-#17 ; 则将 Z 值转换为增量方式
ENDIF

IF #25 GE 0             ; 如果 Z 值相对当前点不下降
    M99                 ; 则返回
ENDIF

N20 X[#23] Y[#24] M03   ; 移到 XY 孔位，且主轴正转
N30 Z[#17]              ; 移到参考点平面 R
N50 G01 Z[#25]          ; 镗孔
N60 G04 P[#15]          ; 在孔底暂停
N70 M05                 ; 主轴停
```




```
N80 M92                   ; 等待用户干预(等待循环启动)

IF #1165 EQ 99            ; 如果为 G99，即返回参考点 R 平面
    #40=#32+#17           ; 定义返回点为参考点平面 R
ELSE
    IF AR[#26] EQ 0
        #40=#32           ; 定义返回点为初始平面 Z
    ELSE
        #40=#26           ; 定义返回点为初始平面 Z
    ENDIF
ENDIF

WHILE #1022 LT #40        ; 如果程编工件位置 Z 小于返回点
    m92                   ; 等待用户干预(等待循环启动)
ENDW

N90 M03                   ; 主轴正转
M99
;______________________________________________________________
%0089
; 镗孔循环 G89 宏程序实现源代码。调用本程序之后，主轴将保持正转

IF AR[#25] EQ 0       ; 没有定义孔底 Z 平面
    M99               ; 则返回
ENDIF

N10 G91               ;用增量方式编写宏程序

IF AR[#23] EQ 90          ; 如果 X 值是绝对方式 G90
    #23=#23-#30       ; 则将 X 值转换为增量方式
ENDIF

IF AR[#24] EQ 90          ; 如果 Y 值是绝对方式 G90
    #24=#24-#31       ; 则将 Y 值转换为增量方式
ENDIF
```



```
IF AR[#17] EQ 90          ; 如果参考点平面 R 值是绝对方式 G90
    #17=#17-#32        ; 则将 R 值转换为增量方式
ELSE
    IF AR[#26] NE 0       ; 初始 Z 平面模态值存在
        #17=#17+#26-#32 ; 则将 R 值转换为增量方式
    ENDIF
ENDIF

IF AR[#25] EQ 90          ; 如果孔底 Z 值是绝对方式 G90
    #25=#25-#32-#17 ; 则将 Z 值转换为增量方式
ENDIF

IF #25 GE 0            ; 如果 Z 值相对当前点不下降
    M99                ; 则返回
ENDIF

N20 X[#23] Y[#24] M03    ; 移到 XY 孔位，且主轴正转
N30 Z[#17]               ; 移到参考点平面 R

N50 G01 Z[#25]           ; 孔加工
N60 M05                  ; 在孔底主轴停
N70 G04 P[#15]           ; 暂停

IF #1165 EQ 99           ; 如果第 15 组 G 代码模态值为 G99
    N90 G00 Z[-#25]          ; 则返回参考点 R 平面
ELSE
    IF AR[#26] EQ 0
        N90 G00 Z[-#25-#17] ;否则返回初始平面
    ELSE
        N90 G90 G00 Z[#26]  ;否则返回初始平面
    ENDIF
ENDIF

N95 M03                  ; 主轴正转
M99
```